Four Stemmers and a Funeral: Stemming in Hungarian at CLEF 2005

Anna Tordai[1] and Maarten de Rijke[2]

[1] Informatics Institute, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam
atordai
[2] mdr@science.uva.nl

Abstract. We developed algorithmic stemmers for Hungarian and used them for the ad-hoc monolingual task for CLEF 2005. Our goal was to determine what degree of stemming is the most effective. Although on average the stemmers did not perform as well as the the best n-gram, we found that stemming over a broad range of suffixes especially on nouns is highly useful.

1 Introduction

In our participation in the CLEF ad-hoc task this year, we focused exclusively on monolingual retrieval for Hungarian. This is the first year Hungarian is part of CLEF, and it is an ideal opportunity to test our work on the effects of stemming in Hungarian. Previous work on languages that are morphologically richer than English, such as Finnish, indicate that there should be benefits from morphological analysis such as stemming, lemmatization, and compound analysis [4, 5, 6]. We have developed a number of suffix-stripping algorithms of varying impact, all focusing on inflectional suffixes. Our goal is to determine the degree of stemming that would prove beneficial for retrieval effectiveness in terms of both precision and recall. We expect to see improvements in recall for all stemmers, but in addition, we hope that our "light" stemmers keep precision at an acceptable level. The "heavy" stemmer we developed is also expected to improve recall, but hurt precision.

The paper is organized as follows. Section 2 describes the traits of the Hungarian language that are important from an information retrieval point of view. Section 3 contains a description of the algorithmic stemmers along with an evaluation. Section 4 describes the retrieval system we used. Section 5 concerns the experiments we performed, finally followed by a conclusion in Section 6.

2 Hungarian Morphology

Hungarian is an agglutinative language remotely related to Finnish and Estonian, and a member of the Ob-Ugric languages [8]. The Hungarian language is highly inflectional, rich in compound words, and has an extensive inflectional

C. Peters et al. (Eds.): CLEF 2005, LNCS 4022, pp. 179–186, 2006.

and derivational morphology. To illustrate this, nouns have 16 to 24 cases depending on the classification system. By adding person, number and possession, a single noun may have as many as 1400 forms [3]. Adjectives similarly may have around 2700 different forms. Verbs have fewer forms, with person, number, tense and transitivity adding up to 59. These numbers merely illustrate the inflectional variety of the language. Additionally, there is an extensive system of derivational suffixes, many of them changing the part of speech of a word.

Compound words are frequent in Hungarian, presenting an additional challenge for retrieval. Compound nouns can be formed by two nouns or a participle and a noun. Adjectives can also be formed by the combination of a noun and adjective. Compounding was not addressed at this time.

3 Algorithmic Stemmers

In this section we describe and evaluate the stemmers used in our retrieval experiments.

3.1 Description of the Stemmers

The stemmers were built in the Snowball language [11] and are rule-based stemmers focusing on inflectional suffixes in Hungarian. Using the Szeged Corpus [1], which is a collection of annotated texts ranging from novels, children's essays, legal texts, newspaper articles to computer books, we created a list of the most frequent types of morphosyntactic tags. This helped to determine which suffixes appear most often in the text and guided the construction of the stemmers.

We developed four types of stemmers:

- *Light1* – handling frequent noun cases, plural and frequent owners.
- *Light2* – handling all noun cases, plural and frequent owners.
- *Medium* – handling frequent noun cases, plural, frequent owners and frequent verb tenses.
- *Heavy* – handling most inflectional suffixes.

The lightest stemmer, *Light1*, only handles 14 frequent noun cases, plural and the most frequent possessive cases. It is the least invasive stemmer but we think it will still have a significant impact. Of all the nouns in the Szeged corpus 26% were in uninflected form. The most frequent types of suffixes cover 36% of the nouns. These were the ones targeted by *Light1* with the exception of the single letter suffix 'k' indicating plurality. Even without it, at least half of all nouns should be indexed in their stem form. Since adjectives have the same case, number and possession suffixes as nouns, they also become stemmed along with numerals which also share a number of cases with nouns.

The second stemmer, *Light2*, is similar to *Light1* except it handles 21 noun cases instead of just 14. Also removing single letter suffixes such as the accusative 't' and superessive 'n'. The *Light1* and *Light2* stemmers both take word length

into account, making sure the remainder is at least a valid vowel-consonant combination.

The third stemmer, *Medium*, removes 12 frequent noun cases, plural, possession and combinations of ownership and plurality. It also handles frequent verb tense-person-number combinations as well as the degree of adjectives. Suffixes forming ordinals and fractions out of numerals were removed.

The last stemmer, *Heavy*, is the most aggressive, removing 21 noun cases, handling plurality and possession. For verbs it handles infinitive, indicative, conditional and subjunctive moods.

3.2 Evaluating the Stemming Algorithms

The stemmers were evaluated both intrinsically and extrinsically. For the intrinsic evaluation, we used Paice's method based on error counting [9]. According to this method, two values determine the quality of a stemmer: *understemming* and *overstemming*. In order to determine these values, a list of words is separated into conceptual groups formed by semantically and morphologically related words. This is the target, and an ideal stemmer should conflate words to these conceptual groups.

The stemmers were used to stem the word list, and following the Paice method their correspondence to the conceptual groups was measured. This resulted in an understemming (UI) and overstemming measure (OI). To determine the general relative accuracy of the stemmers, we use a measure, called *error rate relative to truncation*, or ERRT. It is useful for deciding on the best overall stemmer in cases where one stemmer is better in terms of understemming but worse in terms of overstemming. To calculate the ERRT we created a baseline using length truncation by reducing the words in the world list to their n first letters where n was 9, 10, 11 and 12. The overstemming and understemming measure of these truncated lists defines the truncation line. The values of any reasonable stemmers are found between this line and the origin. Figure 1 shows the UI and OI values for each stemmer with the truncation line. Generally, the further the stemmer is from this line, the better it performs on the word lists. By drawing a line that passes through the origin, the datapoint identified by the pair (UI,OI) consisting of the stemmer's understemming and overstemming index, respectively, and that intersects the truncation line, we obtain the distances necessary to calculate the ERRT value of each stemmer. These are the distance from the origin to the stemmer's (UI,OI) divided by the distance from the origin to the intersection with the truncation line. Low overstemming and understemming indexes are the desired feature in a stemmer. Stemmers that are closer to the origin have lower UI and OI values which means the distance is also shorter. The 'best' stemmer would also have the lowest ERRT value compared to the rest.

Table 1 contains the UI, OI and ERRT values for each of the four stemmers used. As expected, *Light1*, being the lightest stemmer, has the highest understemming index, while *Heavy* has the lowest value. The high value for understemming for *Light1* indicates that it leaves many words unstemmed or just understemmed. The reverse is true for the overstemming index. The *Medium*

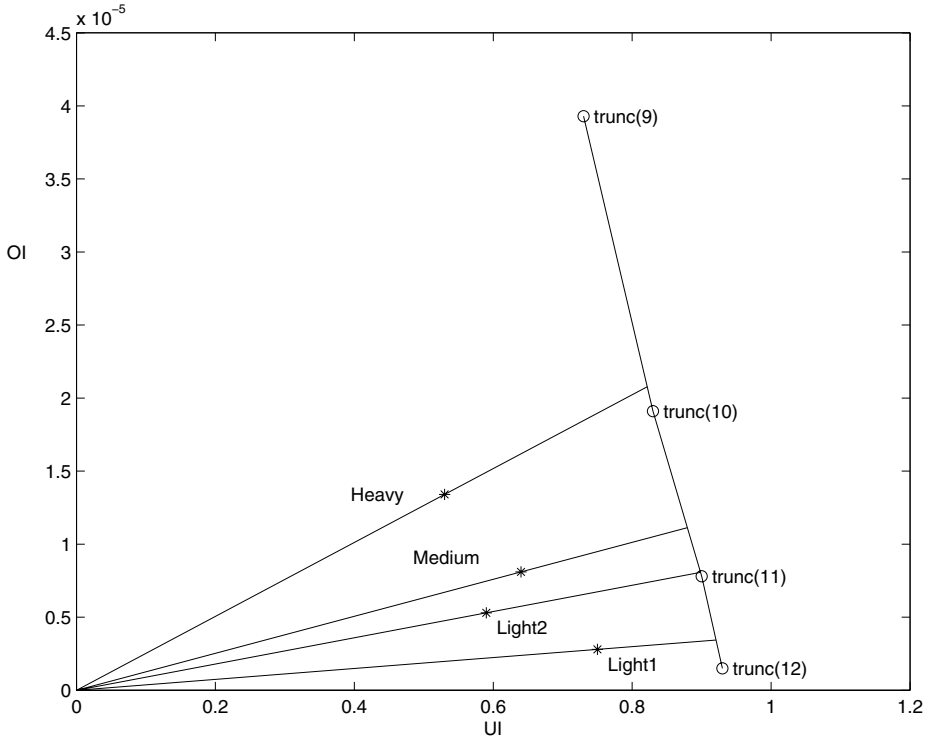

Fig. 1. *UI* × *OI* plot with the *ERRT* distances

stemmer has a lower understemming and higher overstemming index than *Light2* which, at first sight, seems surprising. However, 54% of the words in the list are nouns, and since *Light2* removes all noun cases, just like *Heavy* but unlike *Light1* and *Medium*, these scores make sense. The *Medium* stemmer focuses on some frequent noun cases and verbs. Verbs form only 23% of the word list so the reason for the somewhat unexpected values is simply due to the fact that the *Medium* stemmer stems fewer words than *Light2*. Overall, when it comes to stemming a word list, a stemmer handling all noun cases yields better results than one restricted to the most frequent noun cases and verb tenses. We suspect that this will apply to a lesser extent for retrieval, as words are unique in the word list unlike in a normal corpus.

An examination of the errors in the word list showed that there are difficulties for the stemmers such as overstemming and homonymy. The overstemming of terms such as *nemzet* (nation) to the invalid *nemz* could be alleviated by an exceptions list containing frequent words. Homonymy, for instance with the term *nevet* meaning either 'to laugh' or the accusative form of 'name', can only be solved by looking at the context of the word.

The high ERRT value of *Light1* indicates that although it has very low overstemming it leaves too many words understemmed making it too light. The same

Table 1. Performance of the stemmers on the word-groups

	UI	OI	ERRT
Light1	0.75	0.0000028	0.81
Light2	0.59	0.0000053	0.66
Medium	0.64	0.0000081	0.73
Heavy	0.53	0.0000134	0.65

is true for the *Medium* stemmer, because it focuses on verbs even though there are fewer verbs than nouns in the word list. In this sense, *Light2* and *Heavy* come out as winners having the lowest ERRT values. What would this mean when used in an information retrieval setting? An analysis of English topics used in CLEF 2004 showed that after stopping, over 65% of the words were nouns, only 10% verbs and 12% adjectives. A post submission analysis confirmed these findings for the 2005 Hungarian topics, with 60% of nouns, 23% adjectives and 17% verbs after stopping. Thus, even if a stemmer only concentrates on stemming nouns it should still have an impact on either recall or precision or both. Based on the ERRT values we expect the runs with *Light2* and *Heavy* stemmers to yield a better recall than the other two stemmers and the baseline (no stemming at all). At the same time, precision will probably be negatively affected by the *Heavy* stemmer. These results suggest that the run with *Light2* should have the highest recall and precision values since it has a low understemming ratio and should still stem a large percentage of words.

4 Retrieval Setup

Now that we have described the stemmers, we turn to our retrieval experiments. We used Lucene (off-the-shelf) for indexing and retrieval with a standard vector space model [7]. In addition, we used a stopword list which was created using the Szeged Corpus [1]. We created a list from the 300 most frequent words in the corpus. Numbers and homonyms were removed from the list and it was expanded with pronouns. The result was a list of 188 words.[1] Both the index and queries were stopped. Diacritics were left untouched.

For more information on the ad-hoc track and the collection see [2, 10]. The document collection was encoded in UTF-8. As the Snowball stemmers were created for ISO Latin encoding, the entire collection was converted into ISO Latin 1 encoding without any loss of textual data.

5 The Experimental Results

5.1 Runs

The results of the official CLEF 2005 experiments have been discussed in our Working Notes [12]. We ran the same experiments with some small alterations

[1] The stopword list is available at `http://ilps.science.uva.nl/Resources/`.

such as changes in the stopword list and the separation of hyphenated words. We also performed some new experiments with 4- and 5-grams.

We extended the stopword list with extra terms that appear in practically every query and do not aid retrieval such as *keressünk* (let us search) and *cikk* (article) and their variations. This small change boosted the Mean Average Precision (MAP) and R-precision scores by an average of 0.5.

Additionally we ran experiments with n-grams, this time testing 4-grams and 5-grams. The *4-gram* run returned the highest MAP and R-precision of all the runs.

Analysis of the official runs [12] showed that some relevant documents weren't retrieved because the hyphenated terms in the query and documents were not separated. To this end we performed a new experiment with the best stemmer, Heavy, where we separated hyphenated words in both document collection and queries. The MAP scores and precision scores improved somewhat as a result.

Table 2. Overview of *MAP* scores and *R-precision* scores for the runs. Best scores are in bold face

	MAP	R-prec	% Relevant Docs Retrieved
Light1	0.2245	0.2477	74.7
Light2	0.2911	0.3017	79.1
Medium	0.2417	0.2591	77.2
Heavy	0.2935	0.2921	79.8
Heavy minus hyphen	0.3099	0.3048	83.1
Base	0.1831	0.2096	62.9
4-Gram	**0.3303**	**0.338**	**83.6**
5-Gram	0.3002	0.3057	82.4

Table 2 shows that the *4-gram* has the best performance with respect to MAP, R-precision and number of relevant documents retrieved. Amongst the algorithmic stemmers the *Heavy* stemmer has the highest MAP and R-precision score closely followed by *Light2*. *Medium* scored lower and *Light1* has the worst scores. Overall, when comparing the stemmer scores with the score of the base run, any kind of stemming is better than no stemming at all.

Although the results are to some extent what we had expected, we need to perform a statistical test to determine if there is any significant difference between the methods and stemmers.

We wanted to know if the results of the four different stemming algorithms was significantly different and whether the *4-gram* performed significantly better than the *Heavy* stemmer. A repeated measures ANOVA was performed and showed significant effects for the factor 'stemmer' for both MAP ($F = 12.52$, $df = 5$, $p < 0.01$) and R-precision ($F = 6.99$, $df = 5$, $p < 0.05$); there is a significant difference in the results of the four different stemmers. The results of the *4-gram* however did not differ significantly from the *Heavy* stemmer in both MAP and R-precision.

We examined four queries more closely to find out what the difference is between the performance of the *4-gram* and the *Heavy* stemmer. For the queries

C285 and C298 the *4-gram* outperformed the *Heavy* stemmer. In both cases the queries contained compound words such as *abortuszellenes* (anti-abortion) and *atomerőmű* (nuclear power station). The *4-gram* found the relevant documents containing terms like *abortusz* (abortion) and erőmű (power station) while the *Heavy* run did not.

For the queries C272 and C273 the *Heavy* run outperformed the *4-gram* run. In these cases the queries contained compound words like *kelet-európai* (Eastern European) and *előélete* ('previous life') as well as other frequent words that resulted in the low ranking of the relevant documents by the *4-gram* run.

6 Conclusion

We compared the performance of four different algorithmic stemmers using two forms of evaluation. In Section 3 we found that the *Light2* and *Heavy* stemmers worked best. This has been confirmed by the findings in Section 5 where we also determined that the *Light2* and *Heavy* stemmers worked significantly better for retrieval than *Medium* and *Light1*. This effectively means that stemming nouns and with them adjectives (the two are linked because of similar morphology) is important and makes a difference for retrieval. The stemming of verbs does not seem to have a significant impact.

The *4-gram* had the highest average scores of all the runs, but for this data, it was not significantly higher than the scores of the best stemmer. The *4-gram* has an advantage over our algorithmic stemmers. It is a stemmer and compound splitter all in one. However, as there is no control over what is being 'split' or 'stemmed' this may lead to negative effects on the ranking of the documents when compared to the stemmer.

The next step would be the development of a compound splitter to use in combination with the stemmers. There is also room for improvement on the stemmers themselves, allowing them to handle more irregular forms and increase the number of correct stems.

Acknowledgements

Maarten de Rijke was supported by the Netherlands Organization for Scientific Research (NWO) under project numbers 017.001.190, 220-80-001, 264-70-050, 354-20-005, 612-13-001, 612.000.106, 612.000.207, 612.066.302, 612.069.006.

Bibliography

[1] Szeged Corpus. A morpho-syntactically annotated and POS tagged Hungarian corpus, 2005.

[2] G. M. Di Nunzio, N. Ferro, G. J. F. Jones, and C. Peters. Clef 2005: Ad hoc track overview. URL: `http://www.clef-campaign.org/2005/working_notes/workingnotes2005/dinunzio05.pdf`.

[3] T. Erjavec and M. Monachini. Specifications and notation for lexicon encoding. Technical report, COP Project 106 MULTEXT - East, December 17, 1997.
[4] S. Fissaha Adafre, W.R. van Hage, J. Kamps, G.L. de Melo, and M. de Rijke. The University of Amsterdam at CLEF 2004, 2004.
[5] V. Hollink, J. Kamps, C. Monz, and M. de Rijke. Monolingual document retrieval for European languages, *Information Retrival*, 7:33-52 2004.
[6] T. Korenius, J. Laurikkala, K. Jarvelin, and M. Juhola. Stemming and lemmatization in the clustering of finnish text documents. In *Proceedings of the Thirteenth ACM conference on Information and knowledge management*, 2005, pages 625–633.
[7] Lucene. The Lucene search engine. URL: `http://jakarta.apache.org/lucene/`.
[8] B. Megyesi. The Hungarian language. URL: `http://www.speech.kth.se/~bea/hungarian.pdf`.
[9] C.D. Paice. Method for evaluation of stemming algorithms based on error counting. *Journal of The American Society for Information Science*, 47(8):632–649, 1996.
[10] C. Peters. What happened in clef 2005. URL: `http://www.clef-campaign.org/2005/working_notes/workingnotes2005/peters05.pdf`.
[11] Snowball. The Snowball string processing language. URL: `http://snowball.tartarus.org/`, 2005.
[12] A. Tordai and M. de Rijke. Hungarian monolingual retrieval at clef 2005. 2005. URL: `http://www.clef-campaign.org/2005/working_notes/workingnotes2005/tordai05.pdf`.