Classifying the Hungarian Web

András Kornai
Metacarta Inc.
875 Massachusetts Ave.
Cambridge, MA 02139
andras@kornai.com

Marc Krellenstein
Reed-Elsevier Inc.
200 Wheeler Rd.
Burlington, MA 01803
m.krellenstein@elsevier.com

Michael Mulligan
divine Inc.
1 Wayside Road
Burlington, MA 01803
mulligan@alum.mit.edu

David Twomey
CEHQ, Inc.
145 Rosemary Street Ste H
Needham, MA 02494
dtwomey@theworld.com

Fruzsina Veress
Teragram Corp.
236 Huntington Ave.
Boston, MA 02115
veress@cs.bu.edu

Alec Wysoker
deNovis Inc.
One Cranberry Hill, Suite 203
Lexington, MA 02421
alecw@pobox.com

Abstract

In this paper we present some lessons learned from building vizsla, *the keyword search and topic classification system used on the largest Hungarian portal,* [origo.hu]. *Based on a simple statistical language model, and the large-scale supporting evidence from* vizsla, *we argue that in topic classification only positive evidence matters.*

0 Introduction

Novices are often attracted to menu-based portals because these are easy to navigate. As they get more familiar with the web, users soon realize that their portal covers only a tiny fraction of the web, and move to keyword search engines. But as their information needs and sophistication grow, so does their frustration with simple keyword search. As a result seemingly obscure features, such as boolean searches, wildcards, and topic classification become increasingly relevant to them. To most users, the ideal system would be one that combines the ease of navigation provided e.g. by Yahoo with the near-exhaustive coverage provided e.g. by Google. But topic classification the Yahoo way, by professional editors, is expensive, and the results of using amateur editors, as in dmoz, are often highly questionable.

One way to address the problem of low editorial bandwidth is to automate the topic classification process. Section 1 of this paper describes [origo.hu], a Hungarian portal that uses both manual and automatic topic classification, and gives a brief overview of the keyword search and autoclassification technology developed by Northern Light Technology (NLT, now part of divine Inc) that is deployed on the Hungarian web, which currently has about 20 million unique pages. As we shall see, this is a very successful system, both in terms of standard performance measures and in terms of end-user satisfaction.

In Section 2 we turn to the main question of the paper: why is this algorithm, which is in many ways closer to classic TF-IDF than modern TREC-style topic detection systems, performing so well? We present a formal analysis of what we take to be the essential part of the topic classification problem, and argue that the characteristics revealed by this analysis justify the use of methods that are simpler than generally thought acceptable. We offer our conclusions in Section 3.

1 [origo.hu]

[origo.hu] (the square brackets are part of the branding) is owned and operated by Axelero Inc, the largest Hungarian ISP. It is by far the most popular web portal in Hungary: according to the visitor number statistics published by Median Inc. (see www.webaudit.hu for current numbers), it enjoys the same kind of superiority, being bigger than the next two competitors put together, that the British Navy had when Britannia ruled the waves. The verb *vizslázni* (originally from the noun *vizsla* 'retriever dog', the trademark of the Axelero search engine) entered the Hungarian lan-

guage in the same sense as the verb *to google* is now used in English.

An important measure of user satisfaction, the number of pages downloaded in a single session, is also considerably better for [origo.hu] than its competitors. The independent auditor, Median Inc., defines a single session as no more than 30 minutes inactivity between page downloads: [origo.hu] users need to look at 6.9 pages until they are satisfied, while on the two largest competitors they have to download 7.9 and 8.1 pages respectively. There is currently no obvious way to quantify exactly how much of this effect can be attributed to better search capabilities and relevance ranking, but the conclusion that these play a significant role seems inescapable.

The `vizsla` search bar is placed prominently at the center of the `http://origo.hu` start page. Upon entering a keyword such as *cement* 'id', a results page containing three major results areas is displayed. At the top, we find results from the *katalógus* 'catalog', a Yahoo-style manually filled hierarchical compendium of web pages, in this case showing a search path `agriculture and industry` $\rightarrow$ `building and construction` $\rightarrow$ `construction materials` $\rightarrow$ `adhesives and mortars` $\rightarrow$ `cement`. Upon clicking this last entry, the user gets 10 very high-quality pages, beginning with one discussing the situation of the cement industry in light of the upcoming EU ascension. Below this, we find the URLs and abstracts for the 10 most highly ranked of the 16,684 pages that have the keyword *cement*. Finally, to the left we find a ranked list of NLT-style custom search folders, beginning with `cement, elections,` and `concrete`.[1] If our query is *vízzáró cement* 'water resistant cement' the katalógus is not displayed, the number of pages found is only 303, and the top custom search folders are now `waterproofing, drainage, adhesives-mortars, concrete, surface preparation, bridge con-`

struction, building maintenance, painting and stuccoing, cement, paint industry, and waste management in this order.

The main features of the NLT keyword search engine that distinguish it from competitors, full support of Boolean queries (including full support of negation), phrase search, trailing wildcards, and proximity search, are well known. The page ranking algorithm, which uses links as one of many factors, has been discussed elsewhere (Krellenstein, 2002). Here we concentrate on the topic classification engine, which differs from its TREC counterparts in several relevant respects. First, the number of topics considered is very large (22,000 for the English hierarchy developed at NLT), as opposed to the few dozen to a few hundred topics considered e.g. in the Reuters work. Second, the assumption is that the typical document has only one dominant topic (or none, as we will discuss later). Two-topic documents are rare, three or more topics for a single document occur seldom enough to be negligible in the sense that we see no practical need for returning more than two topics per document (though the engine of course has the facilities for doing so, should the need arise in some non-web application). Finally, we assume that training data is available only in very small quantities, only a handful of documents per category, as opposed to the hundreds of training documents per category used in TREC.

Axelero's `katalógus` system is a mature, highly coherent work of knowledge engineering,[2] with a keyword-spotting hook into the search query system. As such, it provided an excellent basis for the NLT autoclassification system, which was trained on the basis of the high quality *exemplary* documents already manually classified to it. Translating the large NLT topic hierarchy from English to Hungarian was not feasible in the deployment timeframe, but even if it were, we would have been faced with the formidable challenge of finding Hungarian exemplaries for many thousands of highly detailed NLT topics. Using the katalógus also made sense because it was cul-

---

[1] To understand how the elections enter the picture one needs to know that allegations of botched and corrupt privatization of the cement industry were a prominent campaign theme.

[2] The internal coherence of the system no doubt owes a great deal to the fact that originally it was developed by one person, Rudolf Ungváry, Hungarian National Library.

turally more appropriate (e.g. in the selection of sports it has a section for table tennis but not for American football) so the chances of finding more Hungarian webpages on the topic are higher. Besides using a native Hungarian topic hierarchy, the system also relies on a morphological analysis (stemming) component developed specifically for Hungarian by Gábor Prószéky and his associates at Morphologic Inc. We keep both the original (inflected) and the stemmed version available for keyword match and topic classification, since this produces superior results to using either of them alone.

Other than these two instances of necessary localization, there is nothing in our system that is specifically geared toward Hungarian, and therefore we believe that the conclusions we draw about this particular algorithm apply to all topic classification systems with the same broad characteristics:

1. monolingual input
2. small amount of training data available
3. large number of topic categories
4. few documents with multiple topics

In what follows we illustrate some of our points on a version of the old Reuters corpus, keeping the standard (Lewis) test/train split, but removing all articles that have more than one topic, and all topics that have less than three training examples. Needless to say, removal of the multitopic documents and the topics with extremely limited training makes the task easier: Bow TF-IDF (McCallum, 1996) obtains 92.51% correct classification on this set with the default settings. But our intention is not to "report results" on a corpus with 21578 (or, after removal, 8998) documents: our results are on the Hungarian web, a corpus over three orders of magnitude larger, and displaying all the difficulties of real language data, such as lack of consistent style, large numbers of typos, search engine spamming, etc. that are largely absent from Reuters.

2 The bag of words model

We assume a collection of documents D and a system of topics T such that T partitions D into largely disjoint subsets $D_t \subset D (t \in T)$. We will use a finite set of words $w_1, w_2, \ldots, w_N$ arranged on order of decreasing frequency. N is generally in the range $10^5 - 10^6$ – for words not in this set we introduce a catchall *unknown word* w_0. By *general language* we mean a probability distribution G_L that assigns the appropriate frequencies to the w_i either in some large collection of topicless texts, or in a corpus that is appropriately representative of all topics. By the (word unigram) probability model of a topic t we mean a probability distribution G_t that assigns the appropriate frequencies $g_t(w_i)$ to the w_i in a large collection of documents about t. Given a collection C we call the number of documents that contain w the *document frequency* of the word, denoted $DF(w, C)$, and we call the total number of w tokens its *term frequency* in C, denoted $TF(w, C)$.

Assume that the set of topics $T = \{t_1, t_k, \ldots, t_k\}$ is arranged in order of decreasing probability $Q(T) = q_1, q_2, \ldots, q_k$. Let $\sum_{i=1}^{k} q_i = T \leq 1$, so that a document is topicless with probability $q_0 = 1 - T$. The general language probability of a word w can therefore be computed on topicless documents to be $p_w = G_L(w)$ or as $\sum_{i=1}^{k} q_i g_i(w)$. In practice, it is next to impossible to collect a large set of truly topicless documents, so we estimate p_w based on a collection D that we assume to be representative of the distribution Q of topics. It should be noted that this procedure, while workable, is fraught with difficulties, since in general the q_j are not known, and even for very large collections it can't always be assumed that the proportion of documents falling in topic j estimates q_j well.

As we shall see shortly, within a given topic t only a few dozen, or perhaps a few hundred, words are truly characteristic (have $g_t(w)$ significantly higher than the background probability $g_L(w)$) and our goal will be to find these. To this end, we need to first estimate G_L: the trivial method is to use the *uncorrected observed frequency* $g_L(w) = TF(w, C)/L(C)$ where $L(C)$ is the length of the corpus C (total number of word tokens in it). While this is obviously very attractive, the numerical values so obtained tend to be highly unstable. For example, the word *with* makes up about 4.44% of a 55m word sample of the *Wall Street Journal* (WSJ) but 5.00% of a

46m word sample of the *San Jose Mercury News* (Merc). For medium frequency words, the effect is even more marked: for example *uniform* appears 7.65 times per million word in the WSJ and 18.7 times per million in the Merc sample. And for low frequency words, the straightforward estimate very often comes out as 0, which tends to introduce singularities in models based on the estimates.

The same uncorrected estimate, $g_t(w) = TF(w, D_t)/L(D_t)$ is of course available for G_t, but the problems discussed above are made worse by the fact that any topic-specific collection of documents is likely to be orders of magnitude smaller than our overall corpus. Further, if G_t is a Bernoulli source, the probability $P(d|t)$ that a document d containing l_1 instances of w_1, l_2 instances of w_2, etc. is produced by the source for topic t will be given by the multinomial formula

$$\binom{l_0 + l_1 + \ldots + l_N}{l_0, l_1, \ldots, l_n} \prod_{i=0}^{N} g_t(w_i)^{l_i} \qquad (1)$$

which will be zero as long as any of the $g_t(w_i)$ are zero. Therefore, we will *smooth* the probabilities in the topic model by the (uncorrected) probabilities that we obtained for general language, since the latter are of necessity positive. Instead of $g_t(w)$ we will therefore use

$$\alpha g_L(w) + (1 - \alpha)g_t(w) \qquad (2)$$

where α is a small but non-negligible constant, usually between .1 and .3. In the recent literature, e.g. (Zhai and Lafferty, 2001), this is generally called *Jelinek-Mercer smoothing*.[3] There are two ways to justify this method: the trivial one is to say that documents are not fully topical, but can be expected to contain a small α portion of general language. A more interesting justification is to treat the general language probability as a Bayesian prior, the topic-specific frequency as the maximum likelihood estimate based on the observations, so that (2) will be the posterior mean of the unknown probability. For the Reuters experiment, we used the 46m Merc wordcount as our general (background) language model.

[3]Actually the first to apply this technique to topic detection was Gish (1993-1994 Switchboard tasks, see (Colbath, 1998)).

What words, if any, are specific to a few topics in the sense that $P(d \in D_t|w \in d) \gg P(d \in D_t)$? This is well measured by the number of documents containing the word: for example *Fourier* appears in only about 200k documents in a large collection containing over 200m English documents (see www.northernlight.com), while *see* occurs in 42m and *book* in 29m. However in a collection of 13k documents about digital signal processing *Fourier* appears 1100 times, so $P(d \in D_t)$ is about $6.5 \cdot 10^{-5}$ while $P(d \in D_t|w)$ is about $5.5 \cdot 10^{-3}$, two orders of magnitude better. In general, words with low DF values, or what is the same, high IDF (inverse document frequency) values are good candidates for being topic-specific, though this criterion has to be used with care: it is quite possible that a word has high IDF because of deficiencies in the corpus, not because it is inherently very specific. For example, the word *alternately* has even higher IDF than *Fourier*, yet it is hard to imagine any topic that would call for its use more often than others.

Recall that topics are modeled by Bernoulli (word unigram) sources: given a document with word counts l_i and total length n, if we make the naive Bayesian assumption that the l_i are independent, the probability that topic t emitted this document will be obtained by substituting (2) in (1):

$$\binom{l_0 + \ldots + l_N}{l_0, \ldots, l_N} \prod_{i=0}^{N} (\alpha g_L(w_i) + (1 - \alpha)g_t(w_i))^{l_i} \qquad (3)$$

For the 0th topic, general language, (1) and (3) are the same. The log probability quotient $\log P(d|t)/P(d|L)$ of the document being emitted by topic t vs the general language is given by

$$\sum_{i=0}^{N} l_i \log \frac{\alpha g_L(w_i) + (1 - \alpha)g_t(w_i)}{g_L(w_i)} \qquad (4)$$

We rearrange this sum in three parts: where $g_L(w_i)$ is significantly larger than $g_t(w_i)$, when it is about the same, and when it is significantly smaller. In the first part, the numerator is dominated by $\alpha g_L(w_i)$, so we have

$$\log(\alpha) \sum_{g_L(w_i) \gg g_t(w_i)} l_i \qquad (5)$$

which we can think of as the contribution of "negative evidence", words that are significantly sparser for this topic than for general language. In the second part, the quotient is about 1, therefore the logs are about 0, so this whole part can be neglected – words that have about the same frequency in the topic as in general language can't help us distinguish whether the document came from the Bernoulli source associated with the topic t or from the one associated with general language. Note that the summands change sign here in the second part, and as long as the progression of terms is roughly linear, we can extend the limits in both directions without changing the overall zero value.

Finally, the part where the probability of the words is significantly higher than the background probability will contribute the "positive evidence"

$$\sum_{g_L(w_i)<<g_t(w_i)} l_i \log(\alpha + \frac{(1-\alpha)g_t(w_i)}{g_L(w_i)})$$

Since α is a small constant, on the order of .2, while in the interesting cases (such as *Fourier* in DSP vs. in general language) g_t is orders of magnitude larger than g_L, the first term can be neglected and we have, for the positive evidence,

$$\sum_{g_L(w_i)<<g_t(w_i)} l_i(\log(1-\alpha)+\log(g_t(w_i))-\log(g_L(w_i)))$$

In every term the first summand $\log(1 - \alpha)$ is about $-\alpha$. The other two terms $\log(g_t(w_i)) - \log(g_L(w_i))$ measure the (base e) orders of magnitude in frequency over general language: we will call this the *relevance* of word w to topic t and denote it by $r(w, t)$. Some examples of the highest (positive), near-zero, and the lowest (negative) relevances follow:

rank	word	$r(w,\text{alum})$
1	aluminium	13.4176
2	tonnes	12.9357
3	lme	12.0313
4	alumina	11.9061
...		
1185	though	0.0079206
1186	30	0.00377953
1187	under	0.00100579
1188	second	-0.0146792
1189	7	-0.0207462
1190	with	-0.022297
...		
1316	you	-2.20392
1317	name	-2.96474
1318	country	-2.97375
1319	day	-3.03341

Table 1 Samples of r for the alum topic

Since for the positive evidence $-\alpha$ is quite negligible compared to the relevance, positive evidence can be approximated by the more manageable

$$\sum_{g_L(w_i)<<g_t(w_i)} l_i r(w, t) \qquad (6)$$

Needless to say, the real interest is not in determining whether a document belongs to a particular topic s as opposed to general language, but rather in whether it belongs in topic t or topic s. We can compute $\log(P(d|t)/P(d|s))$ as $\log((P(d|t)/P(d|E))/(P(d|s)/P(d|E)))$, and the importance of this step is that we see that the "negative evidence" given by (5) also disappears.

There are two reasons for this. First, the absolute value of the negative evidence is small: on the average Reuters topic, the sum of the negative relevances is less than 5% of the sum of positive relevances. Second, words that are below background probability for topic t will in general be also below background probability for topic s, since their instances are concentrated in some other topic u of which they are truly characteristic. The key contribution in distinguishing topics s and t by computing $\log(P(t|d)/P(s|d))$ will therefore come from those few words that have significantly higher than background probabilities in at least one of these:

$$\sum_{g_L(w_i)<<g_t(w_i)} l_i r(w, t) - \sum_{g_L(w_i)<<g_s(w_i)} l_i r(w, s) \qquad (7)$$

For words w_i that are significant for both topics (such as *Fourier* would be for DSP and for Harmonic Analysis), the contribution of general language cancels out, and we are left with $\sum l_i \log(g_t(w_i)/g_s(w_i))$. But such words are rare even for closely related topics, so the two sums in (7) are largely disjoint.

What (7) defines is the simplest, historically oldest, and best understood pattern classifier, a *linear machine* where the decision boundaries are simply hyperplanes (Highleyman, 1962; Duda et al., 2001). As the above reasoning makes clear, linearity is to some extent a matter of choice: certainly the underlying bag of words assumption, that the words are chosen independent of one another, is quite dubious. However, it is a good first approximation, and one can extend it from Bernoulli (0 order Markov) to first, second, third order, etc. Once the probabilities of word pairs, word triples, etc are explicitly modeled, much of the criticism directed at the bag of words approach loses its grip.[4]

A relevance-based linear classifier containing for all topics all the words that appeared in its training set gives 91.13% correct classification: this has 2154 words in the average topic model. If the least relevant 40% of the words is excluded from the models, average model size decreases to 1454 words, but accuracy actually *improves* to 92.83% (recall that the Bow baseline was 92.51% on this set), demonstrating rather clearly the main thesis that we derived via estimation above, namely that negative and zero evidence is simply noise that we can safely ignore.
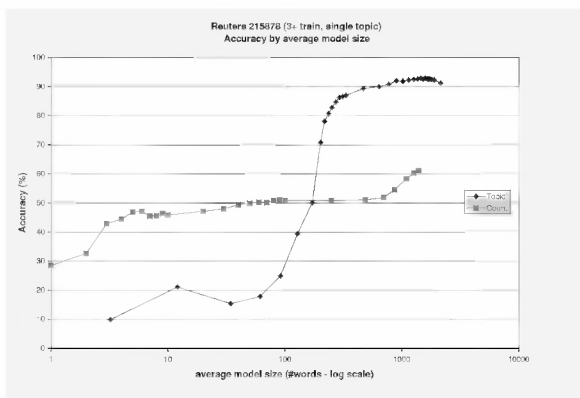
Figure 1. Models with equal number of words vs equal cumulative TF

Figure 1 shows the model-size accuracy tradeoff, with model size plotted on the x axis on a log scale. Note that if we keep only the top 15% of the words (average model size 333), we lose only

---

[4]The NLT system directly indexes word pairs and can match strings of arbitrary length for topic classification.

6.4% of our peak classification performance, since the models still classify 87% correct. If we are prepared to sacrifice another 6% in performance, average model size can be reduced to 236, with classification accuracy still at a very acceptable 80.7% level.

The algorithm used to obtain these numbers simply ranks the words within each model by relevance, and keeps the models balanced by cumulative TF. NLT's proprietary word selection algorithm gets to the 80% level with 30 words per model. Reducing the model size even more drastically would take us out of the realm of practically acceptable classifiers, but as an illustration of our main point it should be noted that keeping the 5 best words in each model would give 46.8% correct classification, and keeping just *one* word, the one with the greatest relevance for each topic, already gives 28.5% correct classification (on this set, random choice would give less than 3%).

3 Conclusions

In Section 2 we argued that for topic classification only positive evidence, i.e. words with significantly higher than background probability, will ever matter. Though we illustrated this point on a standard corpus, we wish to emphasize that it is not this toy example, but rather the objectively measurable user satisfaction with the large-scale system described in Section 1, that provides the empirical underpinnings of our theoretical argument.

If only the best (positive) evidence is used, the models can be *sparse*, in the sense of having nonzero coefficients $r(w, t)$ only for a few dozen, or perhaps a few hundred words w for a given topic t, even though the number of words considered, N, is typically in the hundred thousands to millions (Kornai and Richards, 2002). An important side effect of this approach is that many documents, not containing a sufficient number of keywords for any topic, will be treated as topicless (part of the general language) i.e. they are rejected from classification. Given the nature and quality of many web documents, this is a desirable outcome.

Not knowing that the parameter space is sparse, for $k = 10^4$ topics and $N = 10^6$ words we

would need to estimate $kN = 10^{10}$ parameters even for the simplest (unigram) model. This may be (barely) within the limits of our supercomputing ability, but it is definitely beyond the reliability and representativeness of our data. Over the years, this led to a considerable body of research on *feature selection,* which tries to address the issue by reducing N, and on *hierarchical classification,* which addresses it by reducing k.

We can't discuss here in detail the problems inherent in hierarchical classification, but we note that for a practical topic detection system higher nodes e.g. `film director` are often next to impossible to train, even though lower nodes e.g. `Spielberg, Fellini, ...` will perform well. As for feature selection, we find that much of the literature suffers from what we will call the *once a feature, always a feature* (OAFAAF) fallacy: if a word w is found distinctive for topic t, an attempt is made to estimate $g_s(w)$ for the whole range of s, rather than the one value $g_t(w)$ that we really care about.

The fact that high quality working classifiers such as `vizsla` can be built using only sparse subsets of the whole potential feature set reflects a deep, structural property of the data: at least for the purpose of comparing log emission probabilities across topic models, the G_t can be approximated by sparse distributions S_t. In fact, this structural property is so strong that it is possible to build classifiers that ignore the differences between the numerical values of $g_s(w)$ and $g_t(w)$ entirely, replacing both by a uniform estimate $g(w)$ based on the IDF of w. Traditionally, the l_i multipliers in (7) are known as the term frequency (TF) factor. Such systems, where the classification load is carried entirely by the zero-one decision of using a particular word as a keyword for a topic, are the simplest TF-IDF classifiers, and the estimation method used in Section 2 fits in the broad tradition of deriving IDF-like weights (Robertson and Walker, 1997) from language modeling considerations (?; Hiemstra and Kraaij, 2002; Miller et al., 1999).

What he have done in the body of the paper was to create a new rationale for a classical TF-IDF system, not just for `vizsla` but for any system along the same lines. The notion of *good keywords*

is often used, though not always defined, in information retrieval. We believe that this is an entirely valid notion, and offered a simple operational definition, *has significantly higher than background probability,* to capture it. Our basic claim was that only the good keywords (positive evidence) matter, and the overall performance of our classification system largely supports this assertion.

Acknowledgements

References

S. Colbath Rough'n'Ready: A meeting recorder and browser Perceptual User Interfaces Conference San Francisco, CA, November 1998 220

R.O. Duda, P.E. Hart, and D.G. Stork 2001 *Pattern Classification* John Wiley and Sons

D. Hiemstra and W. Kraaij 1998 TwentyOne at TREC-7: ad-hoc and cross language track *Proceedings of TREC-7* 174–185

W.H. Highleyman 1962 Linear decision functions with application to pattern recognition. *Proceedings of the IRE,* 50:1501–1514.

A. Kornai and J.M. Richards 2002 Linear discriminant text classification in high dimension In: A. Abraham and M. Koeppen (eds): *Hybrid Information Systems* Physica Verlag, Heidelberg 527–538

M. Krellenstein 2002 Operational aspects of the NLT search engine *Proceedings of SIGIR-02*

A.K. McCallum 1996 Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering `http://www.cs.cmu.edu/~mccallum/bow`

D.R. Miller, T. Leek, and R.M. Schwartz 1999 A hidden Markov model information retrieval system *Proceedings of SIGIR-99* 214–221

J.M. Ponte and W.B. Croft 1998 A language modelling approach to information retrieval *Proceedings of SIGIR-98* 275–281

S.E. Robertson and S. Walker 1997 On relevance weights with little relevance information *Proceedings of SIGIR-97* 16–24

Chengxiang Zhai and John Lafferty 2001 A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval *Research and Development in Information Retrieval* 334–342