One Size Fits All?
A Simple Technique to Perform Several NLP Tasks

Daniel Gayo-Avello, Darío Álvarez-Gutiérrez, and José Gayo-Avello

Department of Informatics, University of Oviedo, Calvo Sotelo s/n 33007 Oviedo (SPAIN)
dani@lsi.uniovi.es

Abstract. Word fragments or *n*-grams have been widely used to perform different Natural Language Processing tasks such as information retrieval [1] [2], document categorization [3], automatic summarization [4] or, even, genetic classification of languages [5]. All these techniques share some common aspects such as: (1) documents are mapped to a vector space where *n*-grams are used as coordinates and their relative frequencies as vector weights, (2) many of them compute a context which plays a role similar to stop-word lists, and (3) cosine distance is commonly used for document-to-document and query-to-document comparisons. blindLight is a new approach related to these classical *n*-gram techniques although it introduces two major differences: (1) Relative frequencies are no more used as vector weights but replaced by *n*-gram significances, and (2) cosine distance is abandoned in favor of a new metric inspired by sequence alignment techniques although not so computationally expensive. This new approach can be simultaneously used to perform document categorization and clustering, information retrieval, and text summarization. In this paper we will describe the foundations of such a technique and its application to both a particular categorization problem (i.e., language identification) and information retrieval tasks.

1 Introduction

N-grams are simply text sequences consisting of *n* items, not necessarily contiguous, which can be either words or characters. Frequently, the term *n*-gram refers to slices of adjoining *n* characters including blanks and running over different words. These character *n*-grams are well suited to support a vector space model to map documents. In such a model each document can be considered a *D* dimensional vector of weights, where *D* is the number of unique *n*-grams in the document set while the *i*-th weight in the vector is the relative frequency, within the document to be mapped, for the *i*-th *n*-gram. Thus, having two documents (or a query and a document) a simple similarity measure can be computed as the cosine of the angle between both vectors, this measure is especially interesting because it is not affected by length differences between the compared documents. This approach, exemplified in classical works such as [6] or [7], provides several advantages: it is language independent, quite robust in the face of typographical or grammatical errors and it does not require word-stemming or stop-word removal.

Nevertheless, the *n*-gram vector model has more applications besides information retrieval (i.e., comparing a query with a document). By using the same cosine distance as similarity measure, the model can be applied to document clustering and

J. L. Vicedo et al. (Eds.): EsTAL 2004, LNAI 3230, pp. 267–278, 2004.

categorization [3] [8]. In addition to this, document vectors from similar documents (a cluster or a human-made document set) can be used to obtain a centroid vector [8]. Within this centroid, each *i*-th weight is just the average of the *i*-th weights from all vectors in the set. Such a centroid provides a "context" where performing document comparisons given that it must be subtracted from the document vectors involved in the process. An especially interesting application of *n*-grams where a context had to be provided was Highlights [4] which used vectors to model both documents and these document's "background"[1]. Highlights extracted keywords automatically from a document with regards to its particular background.

Such classical approaches show two major drawbacks: (1) since documents are represented by *D* dimensional vectors of weights, where *D* is the total amount of different *n*-grams in the whole document set, such vectors are not document representations by themselves but representations according to a bigger "contextual" corpus, and (2) cosine similarities between high dimensional vectors tend to be 0 (i.e., two random documents have a high probability of being orthogonal to each other), so, to avoid this "curse of dimensionality" problem it is necessary to reduce the number of features (i.e. *n*-grams), which is usually done by setting arbitrary weight thresholds.

blindLight is a new approach related to those described before and so, applicable to the tasks mentioned above (i.e., document categorization and clustering, information retrieval, keyword extraction or automatic summarization [9]) however it takes into account some important requisites to avoid the problems in previous solutions: First, every document must have assigned a unique document vector with no regards to any corpus and, second, another measure, apart from cosine similarity, has to be used.

2 Foundations of the blindLight Approach

blindLight, as other *n*-gram vector space solutions, maps every document to a vector of weights; however, such document vectors are rather different from classical ones. On one hand, any two document vectors obtained through this technique are not necessarily of equal dimensions, thus, there is no actual "vector space" in this proposal. On the other hand, weights used in these vectors are not relative frequencies but the significance of each *n*-gram within the document.

Computing a measure of the relation between elements inside *n*-grams, and thus the importance of the whole *n*-gram, is a problem with a long history of research, however, we will focus in just a few references. In 1993 Dunning described a method based on likelihood ratio tests to detect keywords and domain-specific terms [10]. However, his technique worked only for word bigrams and were Ferreira da Silva and Pereira Lopes [11] the ones who presented a generalization of different statistical measures so these could be applied to arbitrary length word *n*-grams. In addition to this, they also introduced a new measure, Symmetrical Conditional Probability [12], which overcomes other statistics-based measures. According to Pereira Lopes, their approach obtains better results than those achieved by Dunning.

blindLight implements the technique described by da Silva and Lopes although applied to character *n*-grams rather than word *n*-grams. Thus, it measures the relation among characters inside each *n*-gram and, so, the significance of every *n*-gram, or what is the same, the weight for the components in a document vector.

[1] This background was not a centroid but built using the dataset as just one long document.

With regards to comparisons between vectors, a simple similarity measure such as the cosine distance cannot be straightforward applied when using vectors of different dimension. Of course, it could be considered a temporary vector space of dimension d_1+d_2, being d_1 and d_2 the respective dimensions of the document vectors to be compared, assigning a null weight to one vector's *n*-grams not present in the other and vice versa. However, we consider the absence of a particular *n*-gram within a document rather distinct from its presence with null significance.

Eventually, comparing two vectors with different dimensions can be seen as a pairwise alignment problem. There are two sequences with different lengths and some (or none) elements in common that must be aligned, that is, the highest number of columns of identical pairs must be obtained by only inserting gaps, changing or deleting elements in both sequences.

One of the simplest models of distance for pairwise alignment is the so-called Levenshtein or edit distance [13] which can be defined as the smallest number of insertions, deletions, and substitutions required to change one string into another (e.g. the distance between `"accommodate"` and `"aconmodate"` is 2).

However, there are two noticeable differences between pairwise-alignning text strings and comparing different length vectors, no matter the previous ones can be seen as vectors of characters. First difference is rather important, namely, the order of components is central in pairwise alignment (e.g., DNA analysis or spell checking) while unsuitable within a vector-space model. Second one is also highly significant: although not taking into account the order of the components, "weights" in pairwise alignment are integer values while in vector-space models they are real.

Thus, distance functions for pairwise alignment, although inspiring, cannot be applied to the concerned problem. Instead, a new distance measure is needed and, in fact, two are provided. Classical vector-space based approaches assume that the distance, and so the similarity, between two document vectors is commutative (e.g., cosine distance). blindLight, however, proposes two similarity measures when comparing document vectors. For the sake of clarity, we will called those two documents query (Q) and target (T) although these similarity functions can be equally applied to any pair of documents, not only for information retrieval purposes.

Let Q and T be two blindLight document vectors with dimensions m and n:

$$Q = \left\{ \left(k_{1Q}, w_{1Q}\right) \quad \left(k_{2Q}, w_{2Q}\right) \quad \ldots \quad \left(k_{mQ}, w_{mQ}\right) \right\} \tag{1}$$

$$T = \left\{ \left(k_{1T}, w_{1T}\right) \quad \left(k_{2T}, w_{2T}\right) \quad \ldots \quad \left(k_{nT}, w_{nT}\right) \right\} \tag{2}$$

k_{ij} is the i-th n-gram in document j while w_{ij} is the significance (computed using SCP [12]) for the n-gram k_{ij} within the same document j.

We define the total significance for document vectors Q and T, S_Q and S_T respectively, as:

$$S_Q = \sum_{i=1}^{m} w_{iQ} \tag{3}$$

$$S_T = \sum_{i=1}^{n} w_{iT} \tag{4}$$

Then, the pseudo-alignment operator, Ω, is defined as follows:

$$Q\Omega T = \left\{ (k_x, w_x) \middle/ \begin{array}{l} (k_x = k_{iQ} = k_{jT}) \wedge (w_x = \min(w_{iQ}, w_{jT})), \\ (k_{iQ}, w_{iQ}) \in Q, 0 \le i < m, \\ (k_{jT}, w_{jT}) \in T, 0 \le j < n \end{array} \right\} \quad (5)$$

Similarly to equations 3 and 4 we can define the total significance for $Q\Omega T$:

$$S_{Q\Omega T} = \sum w_{iQ\Omega T} \quad (6)$$

Finally, we can define two similarity measures, one to compare Q vs. T, Π (uppercase Pi), and a second one to compare T vs. Q, P (uppercase Rho), which can be seen analogous to precision and recall measures:

$$\Pi = S_{Q\Omega T} / S_Q \quad (7)$$

$$\mathrm{P} = S_{Q\Omega T} / S_T \quad (8)$$

To clarify these concepts we will show a simple example based on (one of) the shortest stories ever written. We will compare original version of Monterroso's Dinosaur with a Portuguese translation; the first one will play the query role and the second one the target, the *n*-grams will be quad-grams.

Cuando despertó, el dinosaurio todavía estaba allí. (Query)

Quando acordou, o dinossauro ainda estava lá. (Target)

Fig. 1. "El dinosaurio" by Augusto Monterroso, Spanish original and Portuguese translation

In summary, the blindLight technique, although vector-based, does not need a predefined document set where performing NLP tasks and so, such tasks can be achieved over ever-growing document sets or, just the opposite, over just one single document [9]. Relative frequencies are abandoned as vector weights in favor of a measure of the importance of each *n*-gram. In addition to this, similarity measures are analogous to those used in pairwise-alignment although computationally inexpensive and, also, non commutative which allows us to combine both measures, Π and P, into any linear combination to tune it to each NLP task.

The rest of the paper will describe some test bed experiments to evaluate our prototypes at different tasks, namely, language identification, genetic classification of languages, and document retrieval.

3 Language Identification and Genetic Classification of Languages Using blindLight

Natural language identification from digital text has a long tradition and many techniques have been proposed: for instance, looking within the text for particular

Q vector (45 elements)		T vector (39 elements)		QΩT (10 elements)	
Cuan	2.489	va_l	2.545	saur	2.244
l_di	2.392	rdou	2.323	inos	2.177
stab	2.392	stav	2.323	uand	2.119
...		...		_est	2.091
saur	2.313	saur	2.244	dino	2.022
desp	2.313	noss	2.177	_din	2.022
...		...		esta	2.012
ndo_	2.137	a_lá	2.022	ndo_	1.981
nosa	2.137	o_ac	2.022	a_es	1.943
...		...		ando	1.876
ando	2.012	auro	1.908		
avía	1.945	ando	1.876		
_all	1.915	do_a	1.767		

Π: 0.209 P: 0.253

Fig. 2. blindLight document vectors for both documents in Fig. 1 (truncated to show ten elements, blanks have been replaced by underscores). QΩT intersection vector is shown plus Π and P values indicating the similarities between both documents

characters [14], words [15], and, of course, *n*-grams [16] or [17]. Techniques based on *n*-gram vectors using the cosine distance perform quite well in this task. Such techniques usually follow these steps: (1) For each document in the corpus they create an *n*-gram vector, (2) while creating document vectors a centroid vector is also computed, and (3) when an unknown text sample is presented to the system it is (3-a) mapped into the vector space, (3-b) the centroid is subtracted from the sample, and (3-c) compared, by means of the cosine distance, with all the reference documents in the set (which also have had the centroid subtracted). Finally, the reference document found most similar to the sample is used to inform the user in which language the sample is probably written.

The application of blindLight to the construction of a language identifier supposes some subtle differences to previous approaches. On one hand, it is not necessary to subtract any centroid neither from reference documents or the text sample. On the other hand, our language identifier does not need to compare the sample vector to every reference language in the database because a language tree is prepared in advance in order to take the number of comparisons to a minimum.

The language identifier to be described is able to distinguish the following European languages: Basque, Catalan, Danish, Dutch, English, Faroese, Finnish, French, German, Italian, Norwegian, Portuguese, Spanish and Swedish. To assure that the identification is solely made on the basis of the language and is not biased by the contents of the reference documents the whole database consists of literal translations of the same document: the first three chapters of the Book of Genesis.

To build the first version of the language identifier was pretty simple. First, an *n*-gram vector was obtained for every translation of the Genesis. Afterwards, a similarity measure, based on Π and P, was defined, being eventually just Π (being the submitted sample the query). Finally, the identifier only needs to receive a sample of text, to create an *n*-gram vector for that sample and to compute the Π similarity between the sample and each reference document. The highest the value of Π, the most likely the language in the reference to be the one used in the sample.

The second version of the language identifier was inspired by the appealing idea of performing genetic classification of languages (i.e., determining how different human

languages relate to each other) by automatic means. Of course, this idea has yet been explored (e.g., [5], [18], or [19]). However, many of these approaches employ either classical vector-space techniques or Levenshtein distance rightly applied to character or phoneme sequences, so, we found rather challenging the application of blindLight to this problem.

The genetic classification of languages using blindLight was performed over two different linguistic data. The first experiment was conducted on the vectors obtained from the text of the Book of Genesis and, so, produced a tree with 14 languages (Fig. 3). The second experiment, involved vectors computed from phonetic transcriptions of the fable "The North Wind and the Sun" which were mainly obtained from the Handbook of the International Phonetic Association [20]. The languages that took part in this second experiment were: Catalan, Dutch, English, French, Galician, German, Portuguese, Spanish, and Swedish. This task produced a tree with 9 languages (Fig. 4), from which 8 were also present in the results from the first experiment. Both experiments used as similarity measure the expression $0.5\Pi+0.5P$, thus, establishing a commutative similarity measure when comparing languages. A technique similar to Jarvis-Patrick clustering [21] was used to build the dendrograms (Figures 4 and 5), however, describing this technique is out of the scope of this paper.

We are not linguists so we will not attempt to conclude anything from both experiments. Nevertheless, not only both trees are coherent to each other but most of the relations shown in them are also consistent, to the best of our knowledge, with linguistics theories. Even the close relation shown between Catalan and French, both lexically and phonetically, finds support in some authors (e.g., Pere Verdaguer [22]), although it contrasts with those classifications which consider Catalan an Ibero Romance language rather distant from Oïl family (to which French belongs).

The data obtained from the lexical comparison of languages was used to prepare a set of artificial mixed vectors[2], namely, `Catalan-French`, `Danish-Swedish`, `Dutch-German`, `Portuguese-Spanish`, `Italic`, `northGermanic`, and `westGermanic`. Such vectors are simply the Ω-intersection of the different reference vectors belonging to each category (e.g., `westGermanic` involves `Dutch`, `English`, and `German` vectors).

To determine the language in which a sample is written two steps are followed: First, the sample is discriminated against `Basque`, `Finnish`, `Italic`, `northGermanic` and `westGermanic`. Then, depending in the broad category obtained, a second phase of comparisons may be needed. Once these two phases have been completed the system informs the user both about the language and the family to which it belongs.

Although the language identifier needs thorough testing, an extremely simple experiment was performed to get some feedback about its accuracy. 1,500 posts were downloaded from five `soc.culture.*` newsgroups, namely, `basque`, `catalan`, `french`, `galiza`, and `german`.

[2] It can be argued that the described language classification experiments were not needed given that actual language classifications are well-known, at least not to build the language identifier. Nevertheless, such experiments were, in fact, essential because it could not be assumed that artificial language vectors built using blindLight would work as expected by only taking into account data provided by non computational classifications.

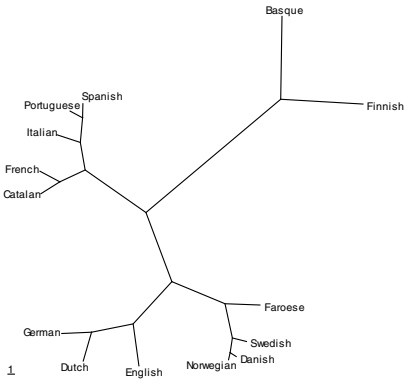

Fig. 3. Unrooted dendrogram showing distances between 14 European written language samples (three first chapters of the Book of Genesis)

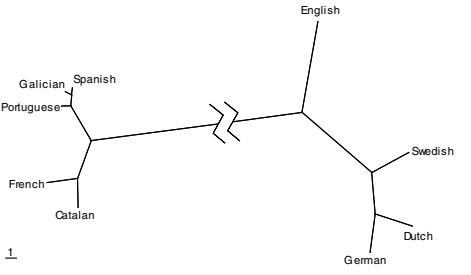

Fig. 4. Unrooted dendrogram showing distances between 9 European oral language samples (phonetic transcriptions of the fable "The North Wind and the Sun"). Distance between Gallo-Iberian (on the left) and Germanic subtrees is 23.985, more than twice the distance shown in the picture

We included, posts from `soc.culture.galiza` to test the system with unknown languages. It must be said that really few posts in that group are actually written in Galician. From those which were actually Galician language samples 63.48% were classified as Portuguese and 36.52% as Spanish, which seems quite reasonable.

Each raw post, without stripping any header information[3], was submitted to the language identifier. Then, if the language assigned by the prototype did not match the supposed language for that post (according to its origin newsgroup) it was human reviewed to check if it was either a system's fault (e.g., assigning English to a post written in any other language), or an actual negative (e.g., a German document posted to `soc.culture.french`). Each fault was added to the count of positives to obtain the total amount of documents written in the target language within its newsgroup and, thus, to compute the language identifier accuracy for that language. Eventually, this was a daunting task because many of these newsgroups suffer from heavy spam and cross-posting problems. The results obtained with this experiment are shown in the following table.

4 Information Retrieval Using blindLight

An information retrieval prototype built upon this approach is participating at the CLEF[4] 2004 campaign at the moment of writing this paper. As with any other application of blindLight, a similarity measure to compare queries and documents is needed. At this moment just two have been tested: Π and a more complex one (see equation 9) which provides rather satisfactory results.

[3] Not stripping the header was done in order to check the system's tolerance to "noise" (i.e., the presence of many English-like text). It was founded that documents with an actual language sample of around 200 characters could be correctly classified in spite of being attached to quite lengthy headers (from 500 to more than 900 characters).

[4] Cross Language Evaluation Forum (`http://www.clef-campaign.org`).

Table 1. Partial results achieved by the language identifier. Accuracy is the fraction of total documents from the newsgroup written in the target language that were correctly identified

Newsgroup	Languages found in the sample posts		Target language	Accuracy
soc.culture.basque	Spanish	96.87%	Basque	100%
	Basque	2.19%		
	English	0.94%		
soc.culture.catalan	Catalan	51.63%	Catalan	98.44%
	Spanish	48.37%		
soc.culture.french	English	73.85%	French	97.56%
	French	25.23%		
	German	0.92%		
soc.culture.german	German	50.35%	German	97.18%
	English	48.94%		
	French	0.71%		

$$\frac{\Pi + norm(\Pi P)}{2} \tag{9}$$

The goal of the *norm* function shown in previous equation is just translate the range of $\Pi \cdot P$ values into the range of Π values, making thus possible a comprehensive combination of both (otherwise, P, and thus $\Pi \cdot P$ values, are negligible when compared to Π).

The operation of the blindLight IR system is really simple:

- For each document in the database an *n*-gram vector is obtained and stored, just in the same way it can be computed to obtain a summary, a list of keyphrases or to determine the language in which it is written.
- When a query is submitted to the system this computes an *n*-gram vector and compares it with every document obtaining Π and P values.
- From these values a ranking measure is worked out, and a reverse ordered list of documents is returned as a response to the query.

This way of operation supposes both advantages and disadvantages: documents may be added to the database at any moment because there is no indexing process; however, comparing a query with every document in the database can be rather time consuming and not feasible with very large datasets. In order to reduce the number of document-to-query comparisons a clustering phase may be done in advance, in a similar way to the language tree used within the language identifier. Of course, by doing this the working over ever-growing datasets is no more possible because the system should be shut down periodically to perform indexing. Thorough performance analysis is needed to determine what database size requires this previous clustering.

There are no yet results about this system's performance at CLEF experiments, however, it was tested on two very small standard collections with promising results. These collections were CACM (3204 documents and 64 queries) and CISI (1460 documents and 112 queries). Both were originally provided with the SMART system[5]

[5] Available at `ftp://ftp.cs.cornell.edu/pub/smart`

and have become a widely used benchmark, thus, enabling comparisons between different IR systems.

Figure 6 shows the interpolated precision-recall graphs for both collections and ranking measures (namely, `pi` and `piro`). Such results are similar to those reached by several systems but not as good as those achieved by other ones; for instance, 11-pt. average precision was 16.73% and 13.41% for CACM and CISI, respectively, while the SMART IR system achieves 37.78% and 19.45% for the same collections. However, it must be said that these experiments were performed over the documents and the queries just as they are, that is, common techniques such as stop-word removal, stemming, or weighting of the query terms (all used by SMART) were not applied to the document set and the queries were provided to the system in a literal fashion[6], as if they were actually submitted by the original users. By avoiding such techniques, the system is totally language independent, at least for non ideographic languages, although performance must be improved.

In addition to this, it was really simple to evolve this system towards cross-language retrieval (i.e., a query written in one language retrieves documents written in another one). This was done without performing machine translation by taking advantage of a sentence aligned corpus of languages source (S) and target (T).

The query written in the source language, Q_s, is splitted in word chunks (from one word to the whole query). The *S* corpus is gathered looking for sentences containing any of these chunks. Every sentence found in *S* is replaced by its counterpart in the *T* corpus. All sentences from *T* corresponding to each chunk within the original query are Ω-intersected. Since such sentences contain, allegedly, the translation of some words from language *S* into language *T*, it can be supposed that the Ω-intersection of their vectors would contain a kind of "translated" *n*-grams (see Fig. 6).

Thus, it is obtained a vector similar, in theory, to that which could be compute from a real translation from the original query. To build this pseudo-translator within the blindLight IR prototype the European Parliament Proceedings Parallel Corpus 1996-2003 [23] has been used obtaining interesting results: in average terms, 38.59% of the *n*-grams from pseudo-translated query vectors are present within the vectors from actual translated queries and, in turn, 28.31% of the *n*-grams from the actual translated query vectors correspond to *n*-grams within the pseudo-translated ones.

5 Conclusions and Future Work

Gayo et al. [9] introduced a new technique, blindLight, claiming it could be used to perform several NLP tasks, such as document clustering and categorization, language identification, information retrieval, keyphrase extraction and automatic summarization from single documents, showing results for these two last tasks.

In this paper the vector model used within blindLight has been refined and the similarity measures used to perform document comparisons have been formalized. In addition to this it has been shown that such a technique can be really applied to language identification, genetic classification of languages and cross-language IR.

---

[6] Just an example query from the CACM collection: `#64 List all articles on EL1 and ECL (EL1 may be given as EL/1; I don't remember how they did it.` The blindLight IR prototype processes queries like this one in an "as is" manner.

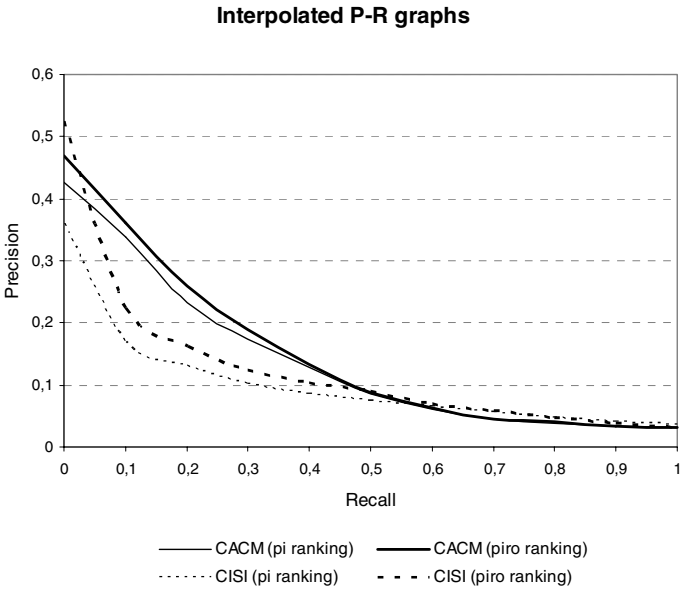

Fig. 5. Interpolated precision-recall graphs for the blindLight IR system applied to CACM and CISI test collections. Top-10 average precision for CACM and CISI was 19.8% and 19.6% respectively, in both cases using `piro` ranking

Query written in language S (from CLEF 2004 French topic list)

Trouver des documents évoquant des discussions sur la réforme des institutions financières, en particulier la Banque Mondiale et le Fond Monétaire International, lors du sommet du G7 qui a eu lieu à Halifax en 1995.

Some sentences from corpus S (Europarl French)

(0861) …la Conférence intergouvernementale sur la réforme des institutions européennes...
(1104) …l'état des travaux concernant la réforme des institutions, réforme qui...
(5116) …le seul grand défi qui se pose à l'Union est la réforme des institutions de l'UE...

Counterpart sentences from corpus T (Europarl English)

(0861) …The Intergov. Conferenc. to address [...] the reform of the European institutions...
(1104) …the state of progress in the reform of the institutions, which is...
(5116) …the single greatest challenge facing the Union is the reform of the EU institutions...

Pseudo-translated query vector (Ω-intersection of previous T sentences)

(…, _ins, _ref, _the, efor, form, inst, itut, nsti, orm_, refo, stit, the_, tion, titu, tuti, utio, …)

Fig. 6. Procedure to pseudo-translate a query written originally in a source language (in this case French) onto a vector containing appropriate n-grams from the target language (English in this example). Blanks have been replaced by underscores, just one chunk from the query has been pseudo-translated

The partial results obtained for language identification prove that it is a robust technique, showing an accuracy higher than 97% with an information-to-noise ratio around 2/7.

The application of this approach to automatic genetic classification of languages, both to lexical and phonological input, produced data coherent to most of linguistics theories and, besides this, useful to improve the operation of a language identifier built using the very same technique.

The performance achieved when applying blindLight to IR is not as good as some IR systems but close to many others. However, it must be noticed that common techniques such as stop-word removal or stemming are not used. This surely has impacted on performance but, this way, the approach is totally language independent. On the other hand, it has been shown how easily cross-language IR can be implemented by performing pseudo-translation of queries (i.e., queries are not actually translated but parallel corpora is used to obtain a vector containing *n*-grams highly alike to be present in actual translations).

Therefore, an extremely simple technique relying on the mapping of documents to *n*-gram vectors in addition to a metric able to compare different length vectors appears to be flexible enough to be applied to a wide range of NLP tasks showing in all of them adequate performance.

References

1. D'Amore, R., Mah, C.P.: One-time complete indexing of text: Theory and practice. Proc. of SIGIR 1985, pp. 155-164 (1985)
2. Kimbrell, R.E.: Searching for text? Send an n-gram! Byte, 13(5), pp. 297-312 (1988)
3. Damashek, M.: Gauging similarity with n-grams: Language-independent categorization of text. Science, 267, pp. 843-848 (1995)
4. Cohen, J.D.: Highlights: Language and Domain-Independent Automatic Indexing Terms for Abstracting. JASIS, 46(3), pp. 162-174 (1995)
5. Huffman, S.: The Genetic Classification of Languages by n-gram Analysis: A Computational Technique, Ph. D. thesis, Georgetown University (1998)
6. Thomas, T.R.: Document retrieval from a large dataset of free-text descriptions of physician-patient encounters via n-gram analysis. Technical Report LA-UR-93-0020, Los Alamos National Laboratory, Los Alamos, NM (1993)
7. Cavnar, W.B.: Using an n-gram-based document representation with a vector processing retrieval model. In Proc. of TREC-3, pp. 269-277 (1994)
8. Huffman, S.: Acquaintance: Language-Independent Document Categorization by N-Grams. In Proceedings of The Fourth Text REtrieval Conference (1995)
9. Gayo-Avello, D., Álvarez-Gutiérrez, D., Gayo-Avello, J.: Naive Algorithms for Key-phrase Extraction and Text Summarization from a Single Document inspired by the Protein Biosynthesis Process. Proc. of Bio-ADIT 2004, LNCS (2004) In press.
10. Dunning, T.: Accurate methods for the statistics of surprise and coincidence. Computational Linguistics, 19(1), pp. 61-74 (1993)
11. Ferreira da Silva, J., Pereira Lopes, G.: A Local Maxima method and a Fair Dispersion Normalization for extracting multi-word units from corpora. In Proc. of MOL6 (1999)
12. Ferreira da Silva, J., Pereira Lopes, G.: Extracting Multiword Terms from Document Collections. Proc. of VExTAL, Venice, Italy (1999)
13. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals, (English translation from Russian), Soviet Physics Doklady, 10(8), pp. 707-710 (1966).
14. Ziegler, D.: The Automatic Identification of Languages Using Linguistic Recognition Signals. PhD Thesis, State University of New York, Buffalo (1991)

15. Souter, C., Churcher, G., Hayes, J., Johnson, S.: Natural Language Identification using Corpus-based Models. Hermes Journal of Linguistics, Vol. 13, pp. 183-203, Faculty of Modern Languages, Aarhus School of Business, Denmark (1994)
16. Beesley, K.R.: Language Identifier: A Computer Program for Automatic Natural-Language Identification of Online Text. In Language at Crossroads: Proceedings of the 19th Annual Conference of the American Translators Association, pp. 47-54 (1988)
17. Dunning, T.: Statistical identification of language. Technical Report MCCS 94-273, New Mexico State University (1994)
18. Kessler, B.: Computational Dialectology in Irish Gaelic. Dublin: EACL. In: Proceedings of the European Association for Computational Linguistics, pp. 60-67 (1995)
19. Nerbonne, J., Heeringa, W.: Measuring Dialect Distance Phonetically, In John Coleman (ed.) Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology, pp.11-18 (1997)
20. Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet, Cambridge University Press (1999)
21. Jarvis, R.A., Patrick, E.A.: Clustering Using a Similarity Measure Based on Shared Near Neighbors, IEEE Transactions on Computers, 22(11), pp. 1025-1034 (1973)
22. Verdaguer, P.: Grammaire de la langue catalane. Les origines de la langue, Curial (1999)
23. Koehn, P.: Europarl: A Multilingual Corpus for Evaluation of Machine Translation, Draft, Unpublished, http://www.isi.edu/~koehn/publications/europarl.ps