INTERNATIONAL JOURNAL OF TRANSLATION
Vol. 19, No. 1

Hierarchical Agglomerative Clustering for Cross-Language Information Retrieval

RAYNER ALFRED[1], ELENA PASKALEVA[2], DIMITAR KAZAKOV[1], MARK BARTLETT[1]
[1]*Computer Science Department, York Univeristy, YORK, UK.*
[2]*Bulgarian Academy of Science, Sofia, Bulgaria.*

ABSTRACT

In this article, we report on our work on applying hierarchical agglomerative clustering (HAC) to a large corpus of documents where each appears both in Bulgarian and English. We cluster these documents for each language and compare the results both with respect to the shape of the tree and content of clusters produced. Clustering multilingual corpora provides us with an insight into the differences between languages when term frequency-based information retrieval (IR) tools are used. It also allows one to use the natural language processing (NLP) and IR tools in one language to implement IR for another language. For instance, in this way, the most relevant articles to be translated from language X to language Y can be selected after studying the clusters of abstracts in language Y.

INTRODUCTION

Effective and efficient document clustering algorithms play an important role in providing intuitive navigation and browsing mechanisms by categorizing large amounts of information into a small number of meaningful clusters. In particular, clustering algorithms build illustrative and meaningful hierarchies out of large document collections, and are ideal tools for their interactive visualization and exploration, as they provide data views that are consistent, predictable and contain multiple levels of granularity.

There has been a lot of research in clustering text documents. However, there are few experiments that compare the results of clustering across languages. It is also interesting to examine the impact on clustering when we reduce the set of terms considered in the clustering process to the set of the most descriptive terms taken from

each cluster. Using the reduced set of terms can be attractive for several reasons. Firstly, clustering a corpus based on a set of reduced terms can speed up the process. Secondly, with the reduced set of terms, we can attempt to use a genetic algorithm to tune the weights of terms to users' needs, and subsequently classify unseen examples of documents.

In this paper, we provide the results of clustering parallel corpora of English-Bulgarian texts, looking at the similarities and differences in three main areas: English-Bulgarian cluster mappings, English-Bulgarian tree structures and the lists of terms that are the most representative for each cluster in English and Bulgarian. Additionally, the effect of term reduction on the cluster mappings and the application of a genetic algorithm in tuning the clustering algorithm are examined.

We will first explain some of the background to (1) the vector space model representation of documents, (2) the hierarchical agglomerative clustering method, (3) genetic algorithms and (4) our semi-supervised clustering technique. Next, we describe the experimental design set-up and the experimental results and draw our conclusions.

BACKGROUND

Vector Space Model Representation

In this work, we use the vector space model (Salton & Michael 1986), in which a document is represented as a vector in an n-dimensional space (where n is the number of different words in the collection of documents). Here, documents are categorized by the words they contain and their frequency. Before obtaining the weights for all the terms extracted from these documents, stemming and stopword removal is performed. Stopword removal eliminates unwanted terms (e.g., those from the closed vocabulary) and thus reduces the number of dimensions in the term-space. Once these two steps are completed, the frequency of each term across the corpus is counted and weighted using *term frequency – inverse document frequency* (tf-idf) (Salton & Michael 1986), as described in equation (1).

Weights are assigned to give an indication of the importance of a word in characterizing a document as distinct from the rest of the corpus. In summary, each document is viewed as a vector whose dimensions correspond to words or terms extracted from the document. The component magnitudes of the vector are the tf-idf weights of the terms. In this model, tf-idf, as described in equation (1), is the product

of term frequency tf(t,d), which is the number of times term t occurs in document d, and the inverse document frequency, equation (2), where |D| is the number of documents in the complete collection and df(t) is the number of documents in which term t occurs at least once. To account for documents of different lengths, the length of each document vector is normalized so that it is of unit length (van Rijsbergen 1979).

$$\text{tf-idf} = \text{tf}(t,d) \cdot \text{idf}(t) \tag{1}$$

$$\text{idf}(t) = \log_{10}\left(\frac{|D|}{\text{df}(t)}\right) \tag{2}$$

$$\text{sim}(d_i, d_j) = \frac{(d_i d_j)}{(\|d_i\| \cdot \|dj\|)} \tag{3}$$

$$\text{Precision (C,L)} = \frac{|C \cap L|}{|C|}, C \in C_{ALL}, L \in L_{ALL} \tag{4}$$

$$\text{Purity} = \sum_{C \in C_{ALL}} \frac{|C|}{|D|} \cdot P(C,L) \tag{5}$$

$$\text{Precision (EBM)} = \frac{|C(E) \cap C(B)|}{|C(E)|} \tag{6}$$

$$\text{Precision (BEM)} = \frac{|C(B) \cap C(E)|}{|C(B)|} \tag{7}$$

Hierarchical Agglomerative Clustering

In this work, we concentrate on hierarchical agglomerative clustering. Unlike partitional clustering algorithms that build a hierarchical solution from top to bottom, repeatedly splitting existing clusters, agglomerative algorithms build the solution by initially assigning each document to its own cluster and then repeatedly selecting and merging pairs of clusters, to obtain a single all-inclusive cluster, generating the cluster tree from leaves to root (Zhao & Karypis 2005). The main parameters in agglomerative algorithms are the metric used to compute

the similarity of documents and the method used to determine the pair of clusters to be merged at each step.

In these experiments, the cosine distance, equation (3), is used to compute the similarity between two documents d_i and d_j. This widely utilized document similarity measure becomes one if the documents are identical, and zero if they share no words. The two clusters to merge at each step are found using the average link method. In this scheme, the two clusters to merge are those with the greatest average similarity between the documents in one cluster and those in the other. Given a set of documents D, one can measure how consistent the results of clustering are for each of the languages to which these documents are translated in the following way. The clusters produced for one language are used as 'gold standard', a source of annotation assigning each document in the set D a cluster label L from the list L_{ALL} of all clusters for that language. Clustering in the other language is then carried out and *purity* (Pantel & Lin 2002), equation (5), used to compare each of the resulting clusters $C \in C_{ALL}$ to its closest match among all clusters L_{ALL}. (*Precision* is the probability of a document in cluster C being labelled L. *Purity* is the percentage of correctly clustered documents.)

Genetic Algorithm

A Genetic Algorithm (GA) is a computational abstraction of biological evolution that can be used to some optimization problems (Holland 1975; Goldberg 1989). In its simplest form, a GA is an iterative process applying a series of genetic operators such as *selection*, *crossover* and *mutation* to a population of elements. These elements, called chromosomes, represent possible solutions to the problem. Initially, a random population is created, which represents different points in the search space. An *objective* and *fitness* function is associated with each chromosome that represents the degree of *goodness* of the chromosome. Based on the principle of the survival of the fittest, a few of the chromosomes are selected and each is assigned a number of copies that go into the mating pool. Biologically inspired operators like *crossover* and *mutation* are applied on these strings to yield a new generation of strings. The process of selection, crossover and mutation continues for a fixed number of generations or till a termination condition is satisfied. More details survey of Genetic Algorithms can be found in (Filho et al. 1994).

Semi-Supervised Clustering Algorithm

As a base for our semi-supervised algorithm, we use an unsupervised clustering method combined with a genetic algorithm incorporating a measure of classification accuracy used in decision tree algorithms, the GINI index (Breiman et al. 1984). Here, we examine the clustering algorithm that minimizes some objective function applied to *k*-cluster centers. In our case, we consider the *cluster dispersion* and *cluster purity*. Before the clustering task, each term is assigned with a specific weight that is normalized across all terms. The main objective is to choose the best weights for all terms considered that minimize some measure of cluster dispersion and cluster quality. In out GA algorithm, the fitness function will be the reciprocal of objective function. Typically *cluster dispersion metric* is used, such as the Davies-Bouldin Index (DBI) (Davies & Bouldin, 1979). DBI uses both the within-cluster and between-clusters distances to measure the cluster quality. Let $d_{centroid}(Q_k)$, defined in (8), denotes the centroid distances within cluster Q_k, where $x_i \in Q_k$, N_k is the number of samples in cluster Q_k, c_k is the center of the cluster and $k \leq K$ clusters. Let $d_{between}(Q_k, Q_l)$, defined in (10), denote the distances between clusters Q_k and Q_l, where c_k is the centroid of cluster Q_k and c_l is the centroid of cluster Q_l.

Therefore, given a partition of the N points into K clusters, DBI is defined in (11). This *cluster dispersion* measure can be incorporated into any clustering algorithm to evaluate a particular segmentation of data. The *Gini index* (GI) has been used extensively in the literature to determine the purity of a certain split in decision trees. Clustering using K cluster centers partitions the input space into K regions. Therefore clustering can be considered as a K-nary partition at a particular node in a decision tree, and GI can be applied to determine the purity of such partition (*cluster purity*). In this case, GI of a certain cluster, k, is computed as defined in (12), where n is the number of class, P_{kc} is the number of points belong to c-th class in cluster k and N_k is the total number of points in cluster k.

$$d_{centroid}(Q_k) = \frac{\sum_i \left\| x_i - c_k \right\|}{N_k} \tag{8}$$

$$c_k = 1/N_k \left(\sum_{x_i \in Q_k} x_i \right) \tag{9}$$

$$d_{between}(Q_k, Q_l) = \left\| c_k - c_l \right\| \tag{10}$$

$$\text{DBI} = \frac{1}{K}\sum_{k=1}^{K} \max_{l \neq k} \left\{ \frac{d_{centroid}(Q_k) + d_{centroid}(Q_l)}{d_{between}(Q_k, Q_l)} \right\} \tag{11}$$

$$GiniC_k = 1.0 - \sum_{c=1}^{n} \left(\frac{P_{kc}}{N_k} \right)^2 \tag{12}$$

$$impurity = \frac{\sum_{k=1}^{K} T_{Ck} \cdot GiniC_k}{N} \tag{13}$$

$$\text{f(N,K)} = \text{Cluster Dispersion + Cluster Purity} \tag{14}$$

$$\text{f(N,K)} = DBI + \frac{\sum_{k=1}^{K} T_{Ck} \cdot GiniC_k}{N} \tag{15}$$

Equation (13) represents the impurity of a particular partitioning into K clusters where N is the number of points in the dataset and T_{Ck} is the number of points in cluster k. The smaller the number the better the quality of clustering we have. In order to get a cluster of better quality, we have to minimize the measure of impurity, defined in (13). In general, the objective function is defined in (14), and in our case, it is computed in (15). By minimizing the objective function defined as the sum of the cluster dispersion measure (DBI) and the cluster impurity measure (represented by the second term in (15)), the algorithm becomes *semi-supervised*. We use this expression to reflect the fact that clustering, typically used as an unsupervised learning technique, has now some of its parameters altered to produce results closer to a given 'gold standard'. More specifically, given N points and K clusters, the term weights are modified to maximize the objective function defined in (15).

EXPERIMENTAL DESIGN

There are three main stages in this experiment. (I) In the first stage, we perform the task of clustering parallel corpora of English-Bulgarian

texts. We look at the similarities and differences in three main areas: English-Bulgarian cluster mappings, English vs Bulgarian tree structures and the extracted most representative terms for English and Bulgarian clusters. (II) Next, in the second stage, we perform the task of clustering the English texts based on the reduced set of terms and comparision with the previous results of clustering English texts using all terms. (III) Finally, we apply the genetic algorithm to optimize the weights of terms considered in clustering the English texts.

I. *Clustering Parallel Corpora*

In the first stage of the experiment, there are two parallel corpora (News Briefs and Features), each in two different languages, English and Bulgarian. In both corpora, each English document E corresponds to a Bulgarian document B with the same content, see Table 1. It is worth noting that the Bulgarian texts have a higher number of terms after stemming and stopword removal.

Table 1. Statistics of Document News and Features

Category (Num Docs)	Language	Total Words	Avg. Words	Different Terms
News briefs (1835)	English	279,758	152	8,456
	Bulgarian	288,784	157	15,396
Features (2172)	English	936,795	431	16,866
	Bulgarian	934,955	430	30,309

The process of stemming English corpora is relatively simple due to the low inflectional variability of English. However, for morphologically richer languages, such as Bulgarian, where the impact of stemming is potentially greater, the process of building an accurate algorithm becomes a more challenging task (Nakov 2003). In this experiment, the Bulgarian texts are stemmed by the BulStem algorithm. English documents are stemmed by a simple affix removal algorithm. Figure 1 illustrates the experimental design set up for the first stage of the experiment. The documents in each language are clustered separately according to their categories (News Briefs or Features) using hierarchical agglomerative clustering. The output of each run consists of three elements: a list of terms characterizing the cluster, the cluster members, and the cluster tree for each set of documents. The next section contains a detailed comparison of the results for the two languages looking at each of these elements.

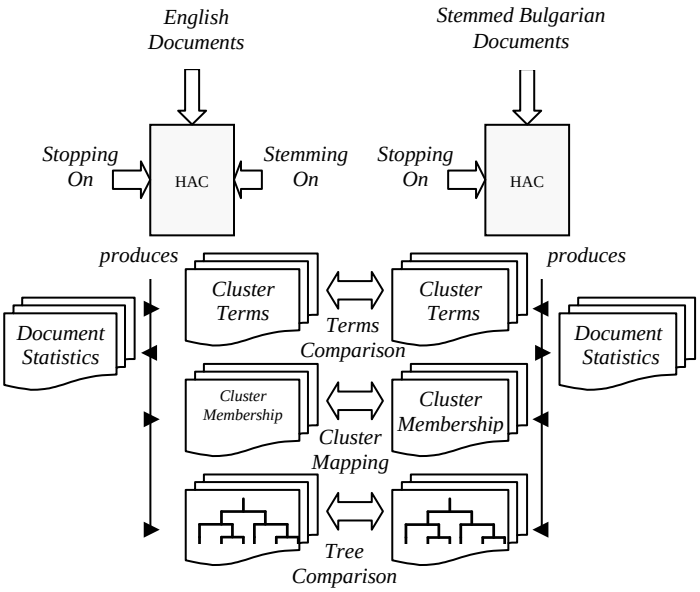

Figure 1. Experimental set up for parallel clustering task

II. *Clustering Document with a Set of Reduced Terms*

In the next stage of the experiment, after clustering the English texts, we examine the terms that characterize the clusters and extract these terms into the set of terms used for clustering the English document again later. We repeat the clustering process for the English texts with only 10, resp. 50 most descriptive terms from each cluster, t, taken from each cluster (k = 10), in which we may have $t \leq 100$ (10 terms from each cluster, k = 10), resp. $t \leq 500$ (50 terms from each cluster, k = 10), due to the fact that the same term may appear in more than one cluster. Figure 2 illustrates the experimental design set up for the second stage of the experiment, in which we repeat the clustering process with a reduced set of terms and compare the results with the previous clustering results.

III. *A Semi-Supervised Clustering Technique Based on Reduced Terms*

The last stage of the experiment uses a corpus where documents are labeled with their target cluster ID. Clustering is then combined with a genetic algorithm optimizing the weight of the terms so that clustering matches as closely as possible the annotation provided. There are two

possible reasons for such an approach. Firstly, one can use the clusters provided for some of the documents in language X as a cluster membership annotation for the same documents in language Y. The additional tuning the GA provides could help cluster the rest of the document in language Y in a way that resembles more closely the result expected if the translation to language X was used. Secondly, experts such as professional reviewers, often produce cluster that are different from the ones generated in an automated way. One can hope that some of their expertise can be captured in the way some of the term weights are modified, and reused subsequently when new documents from the same domain are added for clustering. Here, we describe the representation of the problem in the Genetic Algorithm setting.

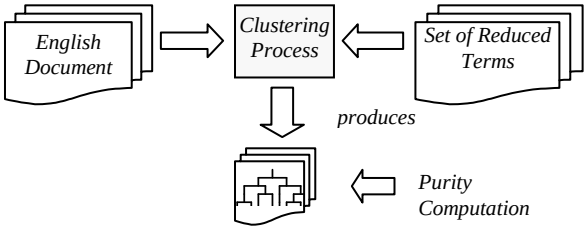

Figure 2. Experimental set up for clustering with the set of reduced terms.

Population Initialization Step: A population of X strings of length m is randomly generated, where m is the number of terms (e.g. cardinality of reduced set of terms). X strings are generated with continuous numbers representing the weight of terms.

Fitness Computation: The computation of the fitness function has two parts: Cluster Dispersion and cluster purity. In order to get clusters of better quality, we need to minimize the DBI, defined in (11). On the other hand, in order to group the same type of objects together in a cluster, we need to minimize the impurity function. Since in GA, we want to maximize the fitness function, the fitness function (OFF) that we want to maximize will be as follows (16).

OFF = 1/Cluster Dispersion + 1/Cluster impurity

$$\text{OFF} = 1/DBI + 1/\left(\frac{\sum_{k=1}^{K} T_{C_k} \cdot GiniC_k}{N}\right) \qquad (16)$$

Selection Process: For the selection process, a roulette wheel with slots sized according to the fitness is used. The construction of such a roulette wheel is as follows:

- Calculate the fitness value f_i, $i \leq X$, for all chromosomes and get the total fitness $T_{Fitness}$ for all X chromosomes.
- Calculate the probability of a selection p_i for each chromosome, $i \leq X$, $p_i = f_i/T_{Fitness}$.
- Calculate the cumulative probability q_i for each chromosome, $q_i = \sum_{j=1}^{i} p_j$.

The selection process is based on spinning the roulette wheel X times: each time we select a single chromosome for a new population in the following way

- Generate a random number r from the range of [0..1].
- Select the i-th chromosome such that $q_{i-1} < r \leq q_i$

Crossover: A pair of chromosomes, c_i and c_j, are chosen for applying the crossover operator with probability p_c. In this experiment, we set p_c = 0.25. This probability gives us the expected number $p_c \cdot X$ of chromosomes that undergo the crossover operation. We proceed by

- Generating a random number r from the range [0..1].
- Performing crossover if $r < p_c$. In this case, for each pair of chromosomes we generate a random integer number *pos* from the range [1..*m-1*] (where m is the length of the chromosome), which indicates the position of the crossing point (i.e., one-point crossover is used).

Mutation: The mutation operator is applied on a bit-by-bit basis. Another parameter of the genetic system, probability of mutation p_m modifies the expected number of mutated bits, equal to $p_m \cdot m \cdot X$. In this experiment, we set p_m = 0.01. For each chromosome and bit within the chromosome, the mutation process:

- Generates a random number of r from the range [0..1].
- Modifies (flips) the bit if $r < p_m$.

As a result of selection, crossover and mutation, the next generation of the population is produced. Its evaluation is used to build the probability distribution for a construction of a roulette wheel with slots sized according to the new fitness values. The rest of the evolution is just a cyclic repetition of selection, crossover, mutation and evaluation until a number of specified generations or specific threshold has been achieved.

EXPERIMENTAL RESULTS

Clustering Parallel Corpora

Mapping of English-Bulgarian Cluster Membership

In the first experiment, every cluster in English is paired with the Bulgarian cluster with which it shares the most documents. The same is repeated in the direction of Bulgarian to English mapping. Two precision values for each pair are then calculated, the precision of the English-Bulgarian mapping (EBM) and that of the Bulgarian-English mapping (BEM). Figures 3–8 show the precisions for the EBM and BEM for the cluster pairings obtained with varying numbers of clusters, k (k = 10, 20, 40) for each of the two domains, News Briefs and Features. The X axis label indicates the ID of the cluster whose nearest match in the other language is sought, while the Y axis indicates the precision of the best match found. For example, in Figure 3, EN cluster 7 is best matched with BG cluster 6 with the EBM mapping precision equal to 58.7% and BEM precision equal to 76.1%.

A final point of interest is the extent to which the EBM mapping matches BEM. When this happens, that is, the best EBM match of BG cluster X is EN cluster Y, and the best BEM match of EN cluster Y is BG cluster X, we say the pair of clusters is aligned. Table 3 shows that alignment between the two sets of clusters is 100% when k = 10 for both domains, News Briefs and Features. However, as the number of clusters increases, there are more clusters that are unaligned. This is probably due to the fact that Bulgarian documents have a greater number of distinct terms. As the Bulgarian language has more word forms to describe English phrases, this may affect the computation of weights for the terms during the clustering process.

Table 2. Purity for Cluster Mapping for English-Bulgarian Documents

Category	k=5	k=10	k=15	k=20	k=40
News briefs	0.82	0.63	0.67	0.65	0.59
Features	N/A	0.77	N/A	0.61	0.54

Table 3. Percentage Cluster Alignment

Category	k = 10	k =20	k = 40
News briefs	100.0%	85.0%	82.5%
Features	100.0%	90.0%	80.0%

It is also possible to study the purity of the mappings. Table 2 indicates the purity of the English-Bulgarian document mapping for various values of k. This measure has only been based on the proportion of clusters that have been aligned, so it is possible to have a case with high purity, but a relatively low number of aligned pairs.

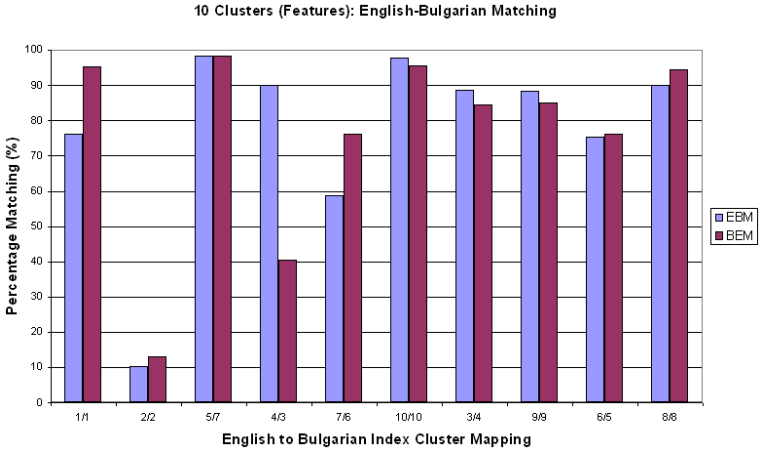

Figure 3. Ten clusters, Features corpus.

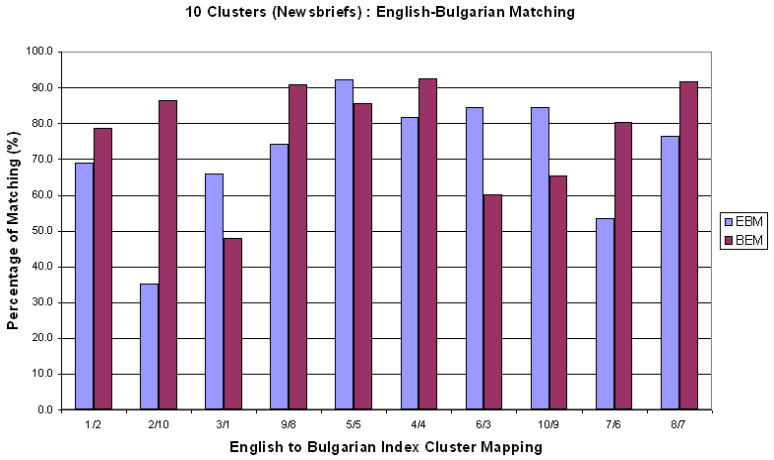

Figure 4. Ten clusters, News Briefs corpus.

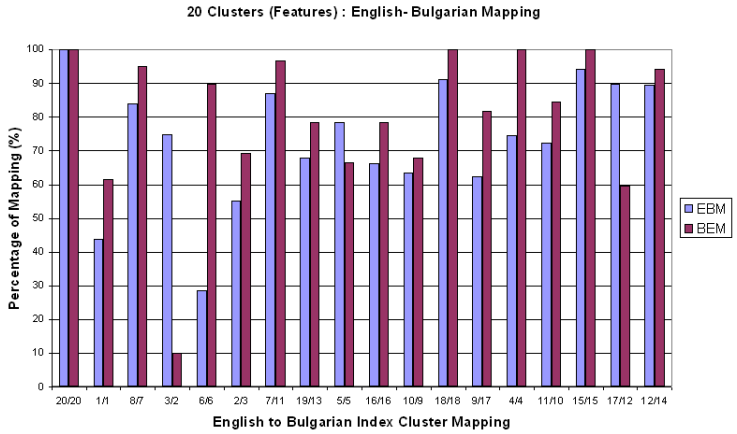

Figure 5. Twenty clusters, Features corpus.

Comparison of HAC Tree Structure

The cluster trees obtained for each language are reduced to a predefined number of clusters (10, 20 or 40) and then the best match is found for each of those clusters in both directions (EBM, BEM). Here, again, we would only pair a Bulgarian cluster C_{BG} with an English cluster C_{EN} if they are each other's best match, that is, C_{BG} —BEM→ C_{EN} and C_{EN} —EBM→ C_{BG}.

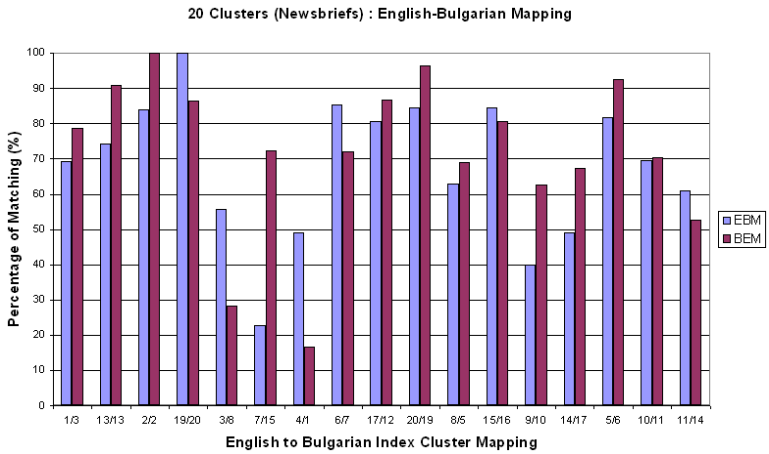

Figure 6. Twenty clusters, News Briefs corpus.

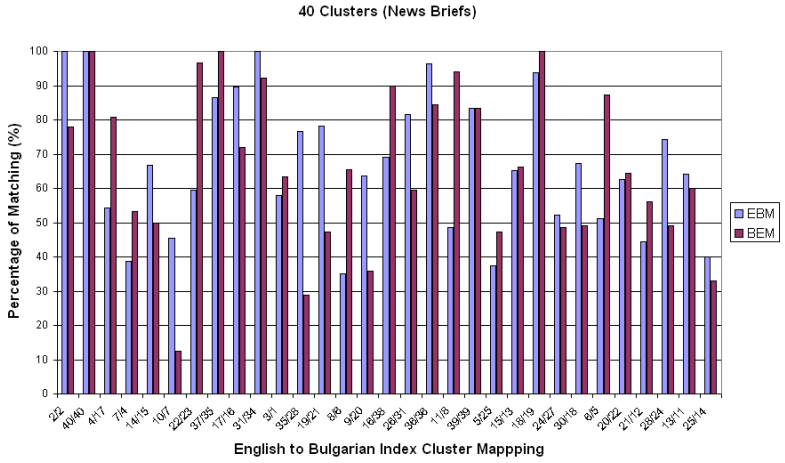

Figure 7: Forty Clusters, News briefs corpus

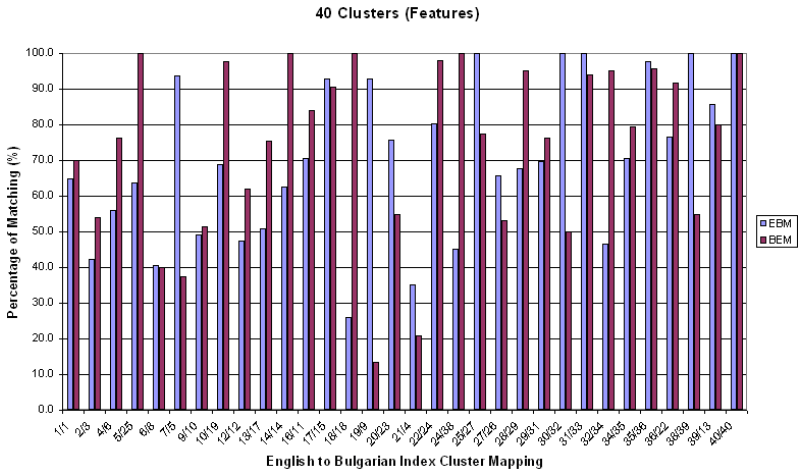

Figure 8. Forty Clusters, Features corpus

The pair of cluster trees obtained for each corpus are compared by first aligning the clusters produced, and then plotting the corresponding tree for each language. Figure 9 and Figure 11 illustrate that when k = 10, all clusters can be paired, and the tree structures for both the English

and Bulgarian documents are identical (although *distances* between clusters may vary). However, when k = 20, there are unpaired clusters in both trees, and after the matched pairs are aligned, it is clear that the two trees are different. We hypothesize that this may be a result of the higher number of stems produced by the Bulgarian stemmer, which demotes the importance of terms that would correspond to a single stem in English.

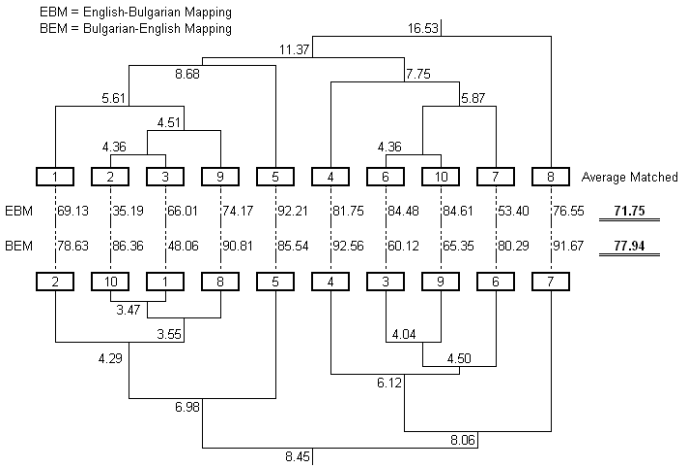

Figure 9. Ten clusters, News Briefs corpus.

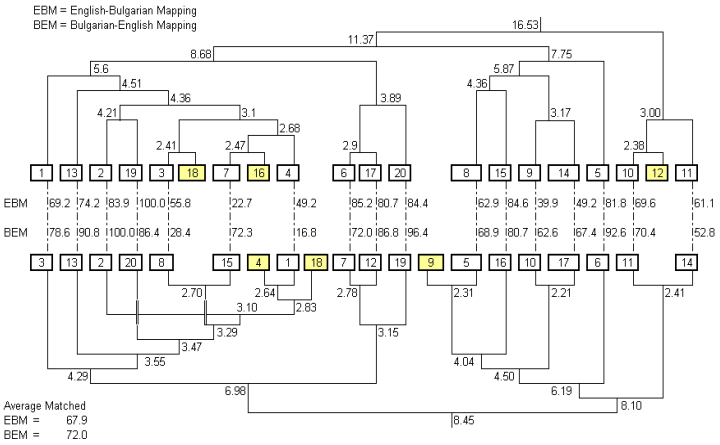

Figure 10. Twenty clusters, News Briefs corpus.

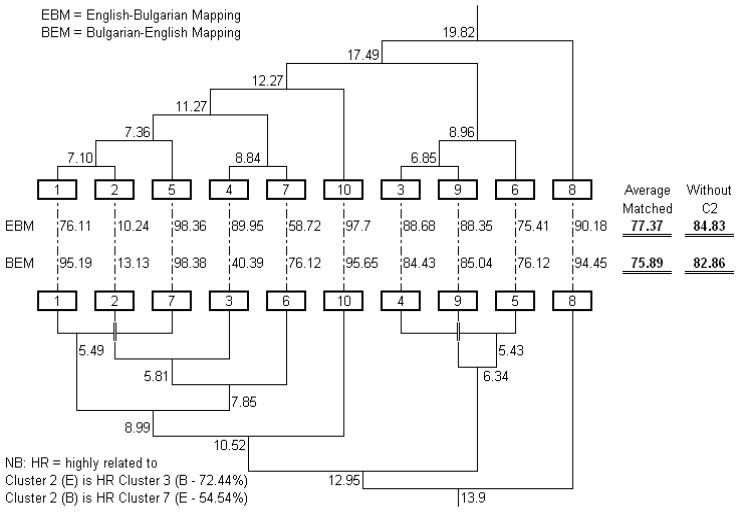

Figure 11. Ten clusters, Features corpus.

Comparison of Terms Extracted

The ten most representative terms that describe the matching English and Bulgarian clusters have a similar meaning as illustrated in Tables 4 and 5. The only notable exception is listed in column 2 of Table 4, where all top Bulgarian terms are related to the topic of 'bird flu', whereas the English terms are split between this topic and the one of 'Olympic games'. This difference disappears when the number of clusters is increased to 20 (and a consistent 'bird flu' $19_{EN}/20_{BG}$ pair of clusters is formed).

Clustering Based on a Set of Reduced Terms

Having seen in the previous experiment that the most representative words for each cluster are similar for each language, an interesting question is whether clustering using only these words improves the overall accuracy of alignment between the clusters in the two languages. The intuition behind this is that, as the words characterizing each cluster are so similar, removing most of the other words from consideration may be more akin to filtering noise from the documents than to losing information.

The clustering is rerun as before, but with only a subset of terms used for the clustering. That is to say, before the tf-idf weights for each document are calculated, the documents are filtered to remove all but *n*

of the terms from them. These n terms are determined by first obtaining 10 clusters for each language, and then extracting the top 10 (resp. 50) terms which best characterize each cluster, with the total number of terms equal to at most $10 \times 10 = 100$ (resp. $10 \times 50 = 500$). Four new sets of clusters are thus created, one for each language and number of terms considered. The results in the four cases are compared to each other, and to the sets of previously obtained sets of clusters for which the full set of terms was used.

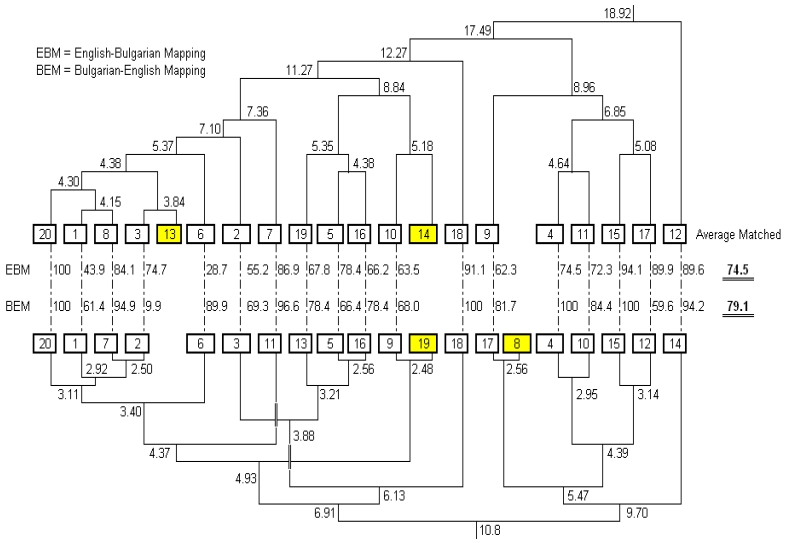

Figure 12. Twenty clusters, Features corpus.

The results of comparing clusters in English and Bulgarian are shown in Table 6. These clearly indicate that as the number of terms used in either language falls, the number of aligned pairs of clusters also decreases. While term reduction in either language decreases the matching between the clusters, the effect is fairly minimal for English and far more pronounced for Bulgarian.

In order to seek to explain this difference between the languages, it is possible to repeat the process of aligning and calculating purity, but using pairs of clusters from the same language, based on datasets with different levels of term reduction. The results of this are summarized in Table 7.

Table 4. Top ten terms for pairs of English and Bulgarian clusters (k = 10, all paired)

C	English	Bulgarian	C
1	macedonia, macedonian, tv, a1, skopj, vesnik, utrinski, makfax, crvenkovski, mia	македони, македонск, А1, цървенковск, скопие, тв, бучковск, утринск, макфакс, трайковск	2
2	olymp, bird, flu, game, test, medal, greek, athen, greec, bronz	грип, птичи, птици, вирус, H5N1, лебед, птичия, случаи, мъртв, щам	10
3	eu, albania, albanian, romania, minist, countri, cent, european, nato, bih	алба, ес, парти, румъни, нато, минист, други, правителств, новин, македони	1
4	kosovo, provinc, statu, unmik, serb, pristina, albanian, belgrad, jessen, petersen	косов, провинци, статут, прищин, юнмик, косовск, йесен-петерсен, оон, сръбск, белград	4
5	turkei, turkish, erdogan, eu, ankara, cypru, cypriot, anadolu, agenc, greek	турци, турск, ердоган, ес, анкар, кипър, анадолск, агенци, кипърск, гюл	5
6	tribun, crime, war, milosev, trial, court, prosecutor, hagu, bosnian, serb	трибунал, престъпл, милошевич, оон, военни, сръбск, обвин, г, хага, понте	3
7	serbia, serbian, montenegro, mladic, belgrad, tanjug, b92, minist, zoran, kostunica	гора, сърбия, Черна, младич, сърбия-Черн, белград, сръбск, б92, танюг, ес	6
8	bih, rs, ashdown, novin, nezavisn, repres, high, republika, srpska, pb	рс, бих, ашдаун, представител, сръбск, независн, новин, пбс, републи, върхов	7
9	bulgarian, bulgaria, mediapool, sofia, btv, iraq, bta, parvanov, minist, trud	българск, българи, ирак, софия, бтв, медиапул, първанов, бнт, бта, минист	8
10	croatia, croatian, gotovina, hina, zagreb, list, sanad, vecernji, ant, hrt	хърват, хърватск, готовин, хина, лист, загреб, санадер, ес, месич, вечер	9

This table demonstrates that, for both languages, as the number of terms considered decreases, the clusters formed deviate further and further from those for the unreduced documents. While the deviation for English is quite low (and may indeed be related to the noise reduction sought), for Bulgarian reducing the number of terms radically alters the clusters formed. As with the earlier experiments, the high morphological variability of Bulgarian compared to English may again be the cause of the results observed.

Semi-Supervised Clustering based on Genetic Algorithm

We have shown that when the language is English, one can reduce the number of terms used without a great loss in performance. This could help reduce the search space and achieve a speed up when the term weights used by a clustering algorithm are fine-tuned by machine learning (e.g. a genetic algorithm) to obtain a tree of clusters in one language that more closely matches the tree for the other language, a novel approach we introduce in (Alfred & Kazakov 2007).

Table 5. Top ten terms for pairs of English and Bulgarian clusters (k = 20 of which 17 paired)

C	English	Bulgarian	C
1	macedonia, macedonian, tv, a1, skopj, vesnik, utrinski, makfax, crvenkovski, mia	македони, македонск, a1, цървенковск, скопие, тв, бучковск, утринск, макфакс, трайковск	3
2	olymp, game, medal, greek, athen, greec, bronz, won, men, stadium	олимпийск, медал, атин, олимпиад, игрит, гърци, спечел, игри, бронзов, категори	2
3	albanian, albania, tirana, osc, elec, moisiu, ata, countri, tuesdai, alfr	алба, нато, македони, ес, албанск, тиран, минист, комиси, европейск, ек	8
4	cent, gt, lt, bih, bank, imf, undp, world, deficit, govern	сръбск, млн, правителств, бежан, други, новин, евро, бих, представител, полици	1
5	kosovo, provinc, statu, unmik, serb, pristina, albanian, belgrad, jessen, petersen	косов, провинци, статут, прищин, юнмик, косовск, йесен-петерсен, оон, сръбск, белград	6
6	turkei, turkish, eu, ankara, erdogan, acces, istanbul, membership, talk, ntv	турци, турск, ес, анкар, ердоган, преговор, членств, кюрдск, нтв, гюл	7
7	eu, romania, romanian, rompr, minist, croatia, european, countri, acces, wednesdai	румъни, румънск, ромпрес, ес, търичану, попеску, найн, о'клок, калин, настас	15
8	tribun, crime, war, milosev, trial, court, prosecutor, hagu, bosnian, serb	трибунал, престъпл, военни, оон, обвин, г, караджич, понте, дел, хага	5
9	serbia, serbian, montenegro, b92, tanjug, djindjic, parti, zoran, belgrad, minist	гора, Черна, сърбия, сърбия-Черн, сръбск, белград, б92, референдум, тадич, танюг	10
10	bih, ashdown, repres, rs, high, novin, nezavisn, reform, pb, ohr	рс, бих, ашдаун, представител, независн, реформ, върхов, новин, пбс, парти	11
11	bih, bosnian, sfor, serb, karadz, rs, srebrenica, search, srpska, republika	рс, сребрениц, сфор, сръбск, босненск, бих, кланет, републи, босненско-сръбск, караджич	14
13	bulgarian, bulgaria, mediapool,	българск, българи, ирак, софия,	13

	sofia, btv, iraq, bta, parvanov, minist, trud	бтв, медиапул, първанов, бнт, бта, минист	
14	mladic, serbia, montenegro, serbian, ratko, ljajic, belgrad, tribun, war, crime	младич, сърбия, гора, Черна, ратко, трибунал, белград, ртс, станкович, сръбск	17
15	croatia, croatian, gotovina, hina, zagreb, list, sanad, vecernji, ant, hrt	хърват, хърватск, готовин, хина, лист, загреб, санадер, ес, месич, вечер	16
17	turkei, iraq, turkish, erdogan, akp, anadolu, billion, gul, recep, tayyip	ирак, турск, турци, пср, ердоган, анадолск, американск, сащ, север, агенци	12
19	bird, flu, h5n1, dead, test, viru, swan, case, strain, found	грип, птичи, птици, вирус, H5N1, лебед, птичия, случаи, мъртв, щам	20
20	cypriot, cypru, turkish, greek, island, plan, reunif, annan, turkei, agenc	кипърск, кипър, остров, турци, плана, гърци, турск, оон, анан, денкташ	19

The comparison of clusters produced from (1) the full term set and (2) the reduced term set for the same language are shown in Table 8. The best weighting scheme found by the GA results in clustering with a lower purity in comparison to the standard tf-idf weighting. This result can be interpreted in the light of several factors. On one hand, the genetic algorithm search is very costly, with 50 generations taking around 5 days on a 1.6GHz Pentium M Dual Core PC with 2GB RAM for a population of 100 chromosomes and a total of 387 terms (top 50 x 10 clusters with some overlap). The main cost of the search is computing the fitness function, i.e., repeating the clustering for each individual in each generation, and evaluating the quality of the result. It seems that the GA simply has not had enough time to find a good solution, which is proved by the fact that, starting from a set of random weights, it has not managed to reproduce the tf-idf baseline performance. On the other hand, the results could also indicate that tf-idf is a very effective weighting scheme, which is, in general, difficult to outperform. A possible answer to both issues would be to bias the initial GA population towards the solutions resembling the standard tf-idf weighting.

CONCLUSION

This paper has presented the idea of using hierarchical agglomerative clustering on a bilingual parallel corpus. The aim has been to illustrate this technique and provide mathematical measures, which can be utilized to quantify the similarity between the clusters in each language. The differences in both clusters and trees (dendrograms) have been

analyzed. We can conclude that with a smaller number of clusters, k, all the clusters from English texts can be mapped into clusters of Bulgarian texts, with a high degree of purity. In contrast, with a larger number of clusters, fewer clusters from English texts can be mapped into the clusters of Bulgarian texts, and the degree of purity decreases, too. In addition, the tree structures for both the English and Bulgarian texts are quite similar when k is reasonable small (and identical for $k \leq 10$).

Table 6. Number of aligned clusters and their purity for reduced term clustering (k = 10)

		Bulgarian Terms		
		All	500	100
English Terms	All	10 (74.9%)	4 (54.2%)	3 (53.0%)
	500	9 (72.9%)	4 (46.0%)	3 (51.5%)
	100	9 (70.3%)	4 (60.1%)	2 (75.5%)

Table 7. Number of aligned clusters and their purity for reduced term datasets against the unreduced datset of the same language (k =10)

	English	Bulgarian
500	10 (74.2%)	4 (53.0%)
100	9 (80.1%)	3 (53.0%)

Table 8. Number of aligned clusters and their purity for reduced term (top 50) datasets against the unreduced datset (k =10)

	English	Bulgarian	English Reduced (top 50)
English Reduced GA (ERGA)	10 (68.1%)	9 (66.0%)	10 (68.0%)

A common factor of all the aspects of parallel clustering studied was the importance that may be attached to the higher degree of

inflection in Bulgarian. From the very beginning, the significantly lower degree of compression that resulted from stemming Bulgarian was noted. This implies that there were a larger number of Bulgarian words which expressed the same meaning, but which were not identified as such. It is likely that this is one of the factors responsible for decreasing the alignment between the clusters for larger values of k. To summarize, here we compared the results of clustering of documents in each of two languages with quite different morphological properties: English, which has a very modest range of inflections, as opposed to Bulgarian with its wealth of verbal, adjectival and nominal word forms. (This difference was additionally emphasized by the fact that the Bulgarian stemmer used produced results which was not entirely consistent in its choice between removing the inflectional or derivational ending.) The clusters produced and the underlying tree structures were compared, and the top 10 most representative terms for each language and cluster listed.

In the paper, we also have also clustered the bilingual English-Bulgarian corpus using a reduced set of terms, and shown the application of a genetic algorithm to tune the weights of terms considered in the clustering process. As most of the top terms seemed to represent the same concepts in the two languages, the possibility of restricting the number of terms used to a much smaller than the original set was considered as a way of making the results more robust with respect to differences between languages and speeding up clustering.

Reducing the number terms alone resulted in a slight decline in performance (a drop of up to 10% in the clusters paired and 4.6% lower cluster purity) when reducing the list of English terms, and a catastrophic decline when this is done for Bulgarian in the cases of 100 and 500 terms studied. When we applied the genetic algorithm to the reduced set of terms to tune the weights of the terms (a maximum of 500 terms) to be considered in the clustering process, the result actually showed a drop in the purity of the clusters. We have already discussed the possible reasons for this and experiments are under way along the lines suggested. Success here would also encourage other possible applications, such as training the algorithm on a hand-clustered set of documents, and subsequently applying it to a superset, including unseen documents, incorporating in this way expert knowledge about the domain in the clustering algorithm.

Clustering multilingual corpora allows one to use the NLP and IR tools in one language to implement IR for another language. For instance, given a collection of Bulgarian research articles for which

an abstract in English exists, one can use the available tools for English to cluster these abstracts, and then request the translation of all articles in the cluster of greatest relevance to a given topic. From another angle, if an overview of the content of a collection of documents in the other language is needed, one could use the English abstracts to cluster the documents, and then translate a sample of documents from each cluster that would be taken as representative of the content of the whole cluster. Of course, any pair of languages can be substituted for the Bulgarian and English used in the example. One of many cases where a similar situation is encountered is in the area of medicine where physicians organized in the {*Cochrane initiative*} review published studies and group them by topic to provide the basis for evidence-based health care (e.g., see [Adams et al. 1998]). Here publications in languages other than English are deemed increasingly important, but the issue of translation represents a major bottleneck that the above mentioned approaches could alleviate.

REFERENCES

Adams C, Duggan L, Wahlbeck K, White P. The Cochrane Schizophrenia Group. Schizophrenia Research 1998;33:185-6.

Alfred, R., and Kazakov, D. 2007. Aggregating Multiple Instances in Relational Databases Using Semi-Supervised Genetic Algorithm-based Clustering. In *the Proc. of MDAI 2007*, Kitakyushu, Japan.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. 1984. *Classification and Regression Trees*. Wadsworth International, California.

Davies, D.L., and Bouldin, D.W. 1979. A cluster separation measure. *IEEE Transactions and Pattern Analysis and Machine Intelligence*, 1(2):224-227.

de Simone, T., and Kazakov, D. 2005. Using WordNet Similarity and Antonymy Relations to Aid Document Retrieval. *RANLP 2005*, Borovets, Bulgaria.

Dumais, S., Landauer, T., and Littman, M. 1996. Automatic cross-linguistic information retrieval using latent semantic indexing. In *SIGIR '96 – Workshop on Cross-Linguistic Information Retrieval*, pp. 16–23.

Filho, J.L.R., Treleaven, P.C., and Alippi, C. 1994. Genetic algorithm programming environments, *IEEE Compu*. 27: 28-43.

Goldberg, D.E. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Inc.

Holland, J. 1975. *Adaption in Natural and Artificial Systems*. Univeristy of Michigan Press.

Hotho, A., Staab, S., and Stumme, G. 2003. *Text clustering based on background knowledge*. Technical Report, No. 425, University of Karlsruhe, Institute AIFB.

Nakov, P. 2003. BulStem: Design and Evaluation of Inflectional Stemmer for Bulgarian. *In Proceedings of Workshop on Balkan Language Resources and Tools* (1st Balkan Conference in Informatics), Thessaloniki, Greece, November.

Pantel, P., and Lin, D. 2002. Document clustering with committees. In *Proc. Of SIGIR'02, Tampere, Finland*.

Salton, G., and Michael, J. 1986. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill Inc., New York, NY.

Sedding, J., and Kazakov, D. 2004. WordNet-based Text Document Clustering. In *Proc. of the 3rd ROMAND workshop*, pp.104-113, Geneva.

van Rijsbergen, C.J. 1979. *Information Retrieval*. Second edition. London: Butterworths.

Zhao, Y., and Karypis, G. 2005. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141.168.

RAYNER ALFRED
PHD CANDIDATE
DEPT. OF COMPUTER SCIENCE
YORK UNIVERSITY, YORK, UNITED KINGDOM
E-MAIL: RALFRED@CS.YORK.AC.UK

DR. ELENA PASKALEVA
ASSOCIATE PROFESSOR
BULGARIAN ACADEMY OF SCIENCE, SOFIA BULGARIA
E-MAIL: < HELLEN@LML.BAS.BG >

DR. DIMITAR KAZAKOV
LECTURER
DEPT. OF COMPUTER SCIENCE
YORK UNIVERSITY, YORK, UNITED KINGDOM
E-MAIL: <KAZAKOV@CS.YORK.AC.UK>

DR. MARK BARTLETT
RESEARCH ASSOCIATE
DEPT. OF COMPUTER SCIENCE
YORK UNIVERSITY, YORK, UNITED KINGDOM
E-MAIL: <BARTLETT@CS.YORK.AC.UK>