Text classification based on phrases

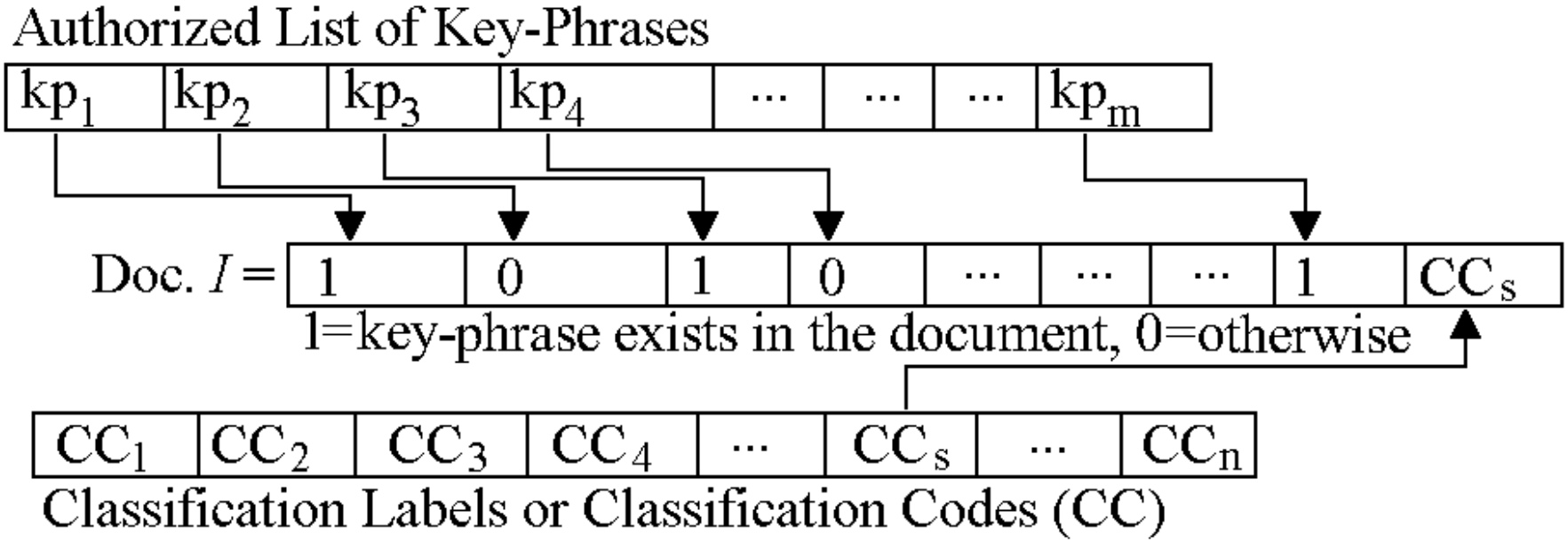
Nikitas N. Karanikolas,
Department of Informatics,
Technological Educational Institute (TEI) of Athens, Greece,
nnk@teiath.gr

International Lectures – May 2016

Overview

- Finding the correct category (class) of a new unclassified document
- Our methodology applies for narrative text
- Two techniques:
  - Based on the distance (similarity) between the new unclassified document and all the pre-classified documents of each class
  - Based on the similarity of the new document to the “Average class document” of each class
- We use key-phrases (text phrases or key terms) as the distinctive features of our text classification methodology
- Our method is based on the automatic extraction of an authority list of key-phrases that is appropriate for discriminating between different classes
- We apply this methodology in handling Greek texts
- Discuss key concepts, algorithms, critical decisions
  (e.g. use of stemming of words in order to produce the key-phrases instead of using key-phrases based on the inflected words).
- A number of parameters of the mining algorithm are also fine-tuned.
- The parameters and the system are evaluated using two training-sets (test collections)
- Useful conclusions are drawn and discussed

representation of existing classified documents (*training set*)



- A predefined (authorized) list of key-phrases
- Each document (of the training set) is represented with a vector where each element of the vector imprints the existence (1) or not (0) of the corresponding authorized key-phrase (kp) in the document.
- The last item of each vector is the classification label (classification code) of the document

Similarity between a new document and the documents of the training set

$$S(D_i, D_{new}) = \frac{\sum_{j=1}^{m} q_j k_{ij}}{\sqrt{\sum_{j=1}^{m} q_j^2 \cdot \sum_{j=1}^{m} k_{ij}^2}} = \frac{\sum_{j=1}^{m} q_j k_{ij}}{\sqrt{\sum_{j=1}^{m} q_j^2} \cdot \sqrt{\sum_{j=1}^{m} k_{ij}^2}} = \frac{\sum_{j=1}^{m} q_j k_{ij}}{L_{D_{new}} \cdot L_{D_i}}$$

- *existing document* D_i is represented by vector $[k_{i1},\ldots,k_{im}]$
- Unclassified document D_{new} is represented by vector $[q_1,\ldots,q_m]$
- m is the number of key-phrases used in the collection

TF - IDF

$k_{ij} = \{0|1\}$

- the weight of key-phrase *j* in some pre-classified document D_i gets the values 0 (not existent) and 1 (existent)
- Since key-phrases (sequences of words in specific order) are selected because they are frequent within the documents of one or few classes but are not so frequent in the documents of the rest classes, even a single existence of some key-phrase in a document is an outstanding mark

$$q_j = \log_2 \left(\frac{ClassCount}{ClassFreq_j} \right)$$

- *ClassCount* is the number of classes of the training set
- $ClassFreq_j$ is the number of classes that include the key-phrase *j*

First methodology: Average similarity with a class

Having the similarities of a new document against any document of the training set, we are able to compute the average similarity of the new document with the classes of the training set. The following function can be used for measuring the average similarity of a new document (D_{new}) with the documents of some class (CL_i)

$$S^{"}(D_{new}, CL_i) = \frac{\sum_{D_j \in DCL_i} S(D_j, D_{new})}{|DCL_i|}$$

- DCL_i is the subset of the training's set documents that are pre-classified as members of class CL_i
- $|DCL_i|$ is the population of documents that constitute DCL_i

Representative key-phrases
Average class document

$$RCL_i = \{kpCL_{i1}, kpCL_{i2}, \ldots, kpCL_{ir}\}$$

- $kpCL_{ij}$ is the *j*-(key-phrase) of the representative key-phrases of class CL_i
- *r* defines the number (the population) of the representative key-phrases for the specified class CL_i ($r = |RCL_i|$)

Second methodology: Similarity with the Average class doc

$$S^{'}(D'_{new}, CL_i) = \frac{\sum_{j=1}^{r} exist(kpCL_{ij}, D'_{new}) \cdot q_{kpCL_{ij}}}{\sqrt{\sum_{f=1}^{|D'_{new}|} 1^2} \cdot \sqrt{\sum_{j=1}^{r} q^2_{kpCL_{ij}}}} = \frac{\sum_{j=1}^{r} exist(kpCL_{ij}, D'_{new}) \cdot q_{kpCL_{ij}}}{\sqrt{|D'_{new}|} \cdot \sqrt{\sum_{j=1}^{r} q^2_{kpCL_{ij}}}}$$

,

$$exist(kpCL_{ij}, D'_{new}) = \begin{cases} 1 & \text{when} \quad kpCL_{ij} \in D'_{new} \\ 0 & \text{when} \quad kpCL_{ij} \notin D'_{new} \end{cases}$$

$$L_{D'_{new}} = \sqrt{|D'_{new}|} \qquad L_{CL_i} = \sqrt{\sum_{j=1}^{r} q^2_{kpCL_{ij}}} \qquad q_{kpCL_{ij}} = \log_2 \left(\frac{ClassCount}{ClassFreq_{kpCL_{ij}}} \right)$$

D'$_{new}$ is a set of key-phrases. It is the subset of authorized key-phrases existing in the new, unclassified, document.

Automatic extraction of an Authority list of key-phrases – wrong way

- Selection of *sequences of words*, which have high frequencies in the documents of the training set. However, *key-phrases* that exist in many documents of the whole training set do not discriminate between documents.
- Selection of *sequences of words* existing in few documents but are quite frequent within them
  - A candidate key-phrase that exists in many documents of only one class (and not in another class) could be erroneously rejected if the number of the documents of this class is greater than the number of documents of other classes.
  - A candidate key-phrase could be erroneously chosen if it exists in a small subset of texts of a "dense" class and all these texts are dedicated on a specific subtopic of the topic of class.
  - The few documents that the candidate key-phrase exists could be "spread" within several classes
- The KEA approach identifies a small number of the document's phrases based on *TF-IDF* and *distance* of the phrase's first occurrence from the beginning of document
  - documents originating from sister classes (classes descent from a common parent) can share the same key-phrases for describing their content but using these common key-phrases does not permit to classify correctly these sister documents

Automatic extraction of an Authority list of key-phrases – suggested way

- Extract key-phrases which are frequent within the documents of one or few classes but are not so frequent in the documents of the rest of the classes of the training set
- Words that constitute key-phrases must always respect distance constraints. They must coexist in a specific window size. The window size is not constant but its size dependents on the number of words that constitute the key-phrase.
- For example the window size for a *2*-word key-phrase could be defined to be 5, while the window size for a *3*-word key-phrase could be defined to be 7

ALCA

Authority List Creation Algorithm

1 For every class (CL_i) of the training set do
2 For every document of the class (DCL_i) do
3 Stemming
4 stopwords removal
5 End {For every document of the class}
6 Choose the most frequent stems of the class (P_0 parameter)
7 Form the candidate double word phrases (C_2) from the frequent stems (L_1)
8 Choose the most frequent double word phrases (L_2) (W_1 and P_1 parameters)
9 For x=3 to mpc do
10 Form the candidate x - width word phrases (C_x) from the frequent (x-1) - width word phrases (L_{x-1})
11 Choose the most frequent x - width word phrases (L_x) (P_{x-1} and W_{x-1} parameters)
12 End {For x=3 to mpc do}
13 Compose an integrated list by joining L_x (for x=2, 3, …, mpc). This join, forms the frequent word phrases of class (FCL_i)
14 End {For every class of the training set}

ALCA

15 Integrate / Join the lists of frequent word phrases of all classes of the training set

16 Reject the frequent word phrases that exist in many classes (P_t parameter). The rest of the frequent word phrases form the set of key-phrases or *Authority List* or *Global Authority List*

17 Form the Dictionary of *Terms*. It is the list of stems that are components of the key-phrases of the *Authority List*.

mpc	maximum number of phrase constituents,
P_0	minimum percentage of texts of the class that must contain a frequent stem,
W_i	width of window that covers *(i+1)*-words phrases, i ∈ [1,2,…,*mpc*-1],
P_i	minimum percentage of texts of the class that must contain a frequent *(i+1)*-words phrase, i ∈ [1,2,…,*mpc*-1],
P_t	maximum percentage of classes that can contain an Authority List's key-phrase.

Forming candidate x-word width key-phrases from frequent (x-1)-word ones

- Step 10 of the ALCA algorithm
- The Cartesian product: *L1 * … * L1* (x-times) is expensive
- The search space can be drastically reduced if larger key-phrases are formed from smaller ones
- It is only necessary to test the occurrences of key-phrases having sub-portions (key-phrases) that are frequent
- Influence by the work on frequent item set algorithms.
- Two phases:
  - The *generation phase* that combines (joins) couples of frequent itemsets of *k* size (couples of L_k members) that have *k-1* common items and produces new candidate itemsets of *k+1* size (candidate for C_{k+1})
  - The *prune phase* that removes such candidate *k+1* size itemsets (candidates from C_{k+1}) that include a *k* size subset that is not a frequent itemset (not a member of L_k)

In ALCA algorithm there is no need for a prune phase

- The frequent itemsets are sets of items without any order
- On the contrary, in the case of frequent key-phrases the order of words is a significant feature
- Moreover the window size varies for different sizes of key-phrases (e.g. window size for *2*-word sequences can be 5, and be 7 for *3*-word sequences)
- Thus, in ALCA algorithm there is no need for a prune phase

Example of no need for prune

Assume now that our data (documents of the processed class) are the following four texts:

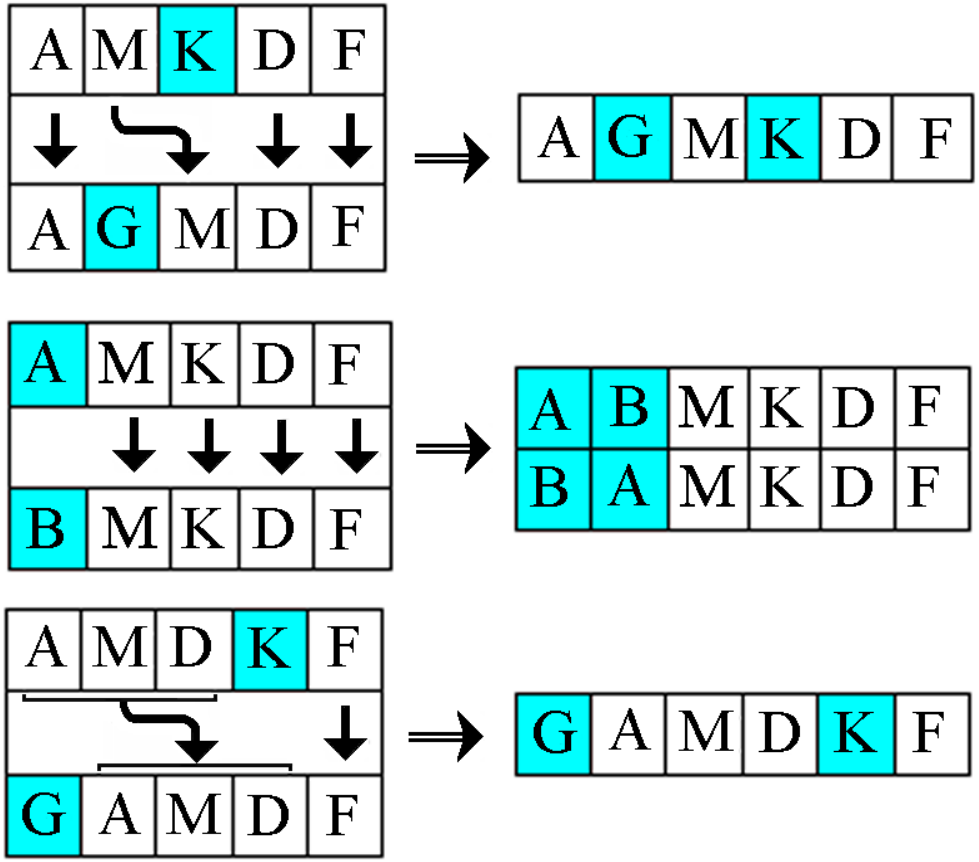
… A ? ? B ? D …
… A ? ? ? B ? D …
… A ? ? ? B D …
… A ? B ? D …

where ? represents a single word that is not one of A, B or D,
… represents one or more words such that none of them is A, B or D.

Assume also that L_2 contains the itemsets AB, AC, BC and BD and that the generation phase suggest (as candidates for C_3) the itemsets ABC, ABD and BCD

If ALCA algorithm used the prune phase then we would have to discard the production ABD, since AD is not included in L_2. This would be wrong since the production ABD has 4 occurrences (it is very frequent) in the window of size 7.

Examples of generation phase for C_6



- The pairs of L_5 that we combine have *x-2* (= 4) common constituents (stems)
- The combination (generation) rule produces a new candidate *6*-word key-phrase by keeping the four common constituent words in the same order and interpolating the unmatched words (according to their original position)

Parameters and concepts under evaluation

- use similarities between a given new unclassified document with **every pre-classified document** of the training set **versus** use similarities between the given new unclassified document with the **Average class documents** for each class
- use of ***(Global) Authority List*** **versus** use of ***Class specific Authority sub-Lists*** for indexing the documents of the training set
- what is the best P_t **parameter** (remind that the ALCA algorithm creates separately the frequent word phrases list for each class (FCL_i), it merges these lists and finally rejects the frequent word phrases that exist in many classes with regard to the P_t parameter)

(Global) Authority List versus Class specific Authority sub-Lists

L_2 of the first class (CL_1)

1	ΑΚΙΝΗΤΟΠΟΙ.ΕΝΟΙΚ.
2	ΑΝΑΛΗΨ.ΑΝΤΙΚ.
3	ΑΝΑΛΗΨ.ΚΑΡΤ.
4	ΑΝΑΛΗΨ.ΧΡΗ.
...	
16	ΑΣΦΑΛ.ΣΥΝΕΡΓ.
17	**ΑΤΟΜ.ΚΑΤΗΓΟΡΟΥ.**
18	ΑΥΤΟΚΙΝ.ΟΠΟΙ.
19	**ΑΥΤΟΚΙΝ.ΣΥΝΕΧ.**
20	ΑΦΑΙΡ.ΑΝΑΛΗΨ.

L_2 of the first class (CL_1) continued

...	
65	ΟΠΟΙ.ΔΙΑΡΡΗΞ.
66	**ΟΠΟΙ.ΣΥΝΕΧ.**
67	ΠΑΡΑΒΙ.ΕΙΣΗΛΘ.
...	
88	ΧΑΡΑΚΤΗΡ.ΤΡΟΠ.
89	ΧΡΗ.ΗΛΕΚΤΡ.
90	ΧΡΗ.ΤΙΜΑΛΦ.

L_3 of the first class (CL_1)

1	ΑΝΑΣΤΑΤ.ΑΦΑΙΡ.ΚΑΡΤ.
2	ΑΝΑΣΤΑΤ.ΣΠΙ.ΑΦΑΙΡ.
3	ΑΝΤΙΚ.ΑΞΙ.ΕΥΡ.
4	ΑΞΙ.ΕΥΡ.ΤΡΕ.
5	ΑΣΦΑΛ.ΑΠΟΥΣ.ΕΝΟΙΚ.
...	
48	ΤΡΟΠ.ΔΡΑ.ΣΥΓΚΕΚΡΙ.
49	ΦΟΡ.ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.
50	ΦΟΡ.ΔΙΕΠΡΑΤ.ΚΛΟΠ.
51	ΧΑΡΑΚΤΗΡ.ΔΡΑ.ΣΥΓΚΕΚΡΙ.
52	ΧΑΡΑΚΤΗΡ.ΤΡΟΠ.ΣΥΓΚΕΚΡΙ.

L_4 of the first class (CL_1)

1	ΑΝΑΣΤΑΤ.ΑΦΑΙΡ.ΚΑΡΤ.ΑΝΑΛΗΨ.
2	ΑΝΤΙΚ.ΑΞΙ.ΕΥΡ.ΤΡΕ.
3	ΑΞΙ.ΕΥΡ.ΤΡΕ.ΔΡΑΣΤ.
4	ΑΞΙ.ΕΥΡ.ΤΡΕ.ΕΙΣΗΛΘ.
5	ΑΤΟΜ.ΚΑΤΗΓΟΡΟΥ.ΚΛΟΠ.ΔΙΑΡΡΗΞ.
...	
22	ΟΜΑ.ΦΟΡ.ΔΙΕΠΡΑΤ.ΚΛΟΠ.
23	ΣΠΙ.ΑΦΑΙΡ.ΚΑΡΤ.ΑΝΑΛΗΨ.
24	ΤΡΕ.ΔΡΑΣΤ.ΠΑΡΑΒΙ.ΕΙΣΗΛΘ.
25	ΦΟΡ.ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.ΚΛΟΠ.
26	ΧΑΡΑΚΤΗΡ.ΤΡΟΠ.ΔΡΑ.ΣΥΓΚΕΚΡΙ.

(Global) Authority List versus Class specific Authority sub-Lists

- *L2* has 90 key-phrases, *L3* has 52 key-phrases and *L4* has 26 key-phrases
- *FCL1* has 168 (= 90 + 52 + 26) key-phrases.
- The populations of the frequent word phrases of the other classes are 24, 14, 113 and 34, respectively.
- There are some key-phrases that exist in more than one class. For example, key-phrase “ΑΤΟΜ.ΚΑΤΗΓΟΡΟΥ.” exists also in *FCL3*, key-phrase “ΑΥΤΟΚΙΝ.ΣΥΝΕΧ.” exists also in *FCL4* and key-phrase “ΟΠΟΙ.ΣΥΝΕΧ.” exists also in *FCL5*.
- There are 9 key-phrases that exist in two classes.
- We can summarize that the discussed training set has
  - 353 (= 168 + 24 + 14 + 113 + 34) key-phrases,
  - 335 of those exist only in one class,
  - 9 of those exist in two classes,
  - there are 344 (= 335 + 9) discrete key-phrases.
- If we decide to reject frequent word phrases that exist in two or more classes (according to step 16 of the ALCA algorithm) then:
  - the (Global) Authority List is reduced to 335 (from 344) items,
  - the Class specific Authority sub-Lists are reduced to have 165, 21, 10, 109 and 30 items, respectively.

(Global) Authority List versus Class specific Authority sub-Lists

- As it is provided earlier, the (Global) Authority List has 335 items and the Class specific Authority sub-Lists (one for each of the five document classes) have 165, 21, 10, 109 and 30 items, respectively.
- Using the Class specific Authority sub-Lists for indexing the pre-classified documents of the training set we actually restrict the method to use only the 165, 21, 10, 109 and 30 key-phrases while indexing the documents of the five classes, respectively.
- Consequently, the extracted lists of representative key-phrases (one for each class) are equivalent to the corresponding Class specific Authority sub-Lists.
- However, using the (Global) Authority List, while indexing any document of the training set, results in finding and using 192, 49, 30, 122 and 47 key-phrases existing in the documents of the first, second, third, fourth and fifth classes, respectively.
- It is possible for some word phrases to be removed (by steps 8 and 11) by some class's frequent word phrases (let say removed from FCL_k). However the same word phrases can be used in some others class's frequent word phrases (let say used in FCL_g) and consequently included in the (Global) Authority List.
- Using the (Global) Authority List for indexing the pre-classified documents of the training set, it is possible to use the early rejected from FCL_k word phrases while indexing the documents of class k.
- This can happen because some documents of class k contain the sparse, for class k, word phrases. Therefore, the use of (Global) Authority List for indexing the pre-classified documents of the training set has the consequence of practically extending the extracted lists of representative key-phrases.

Other parameters – not evaluated

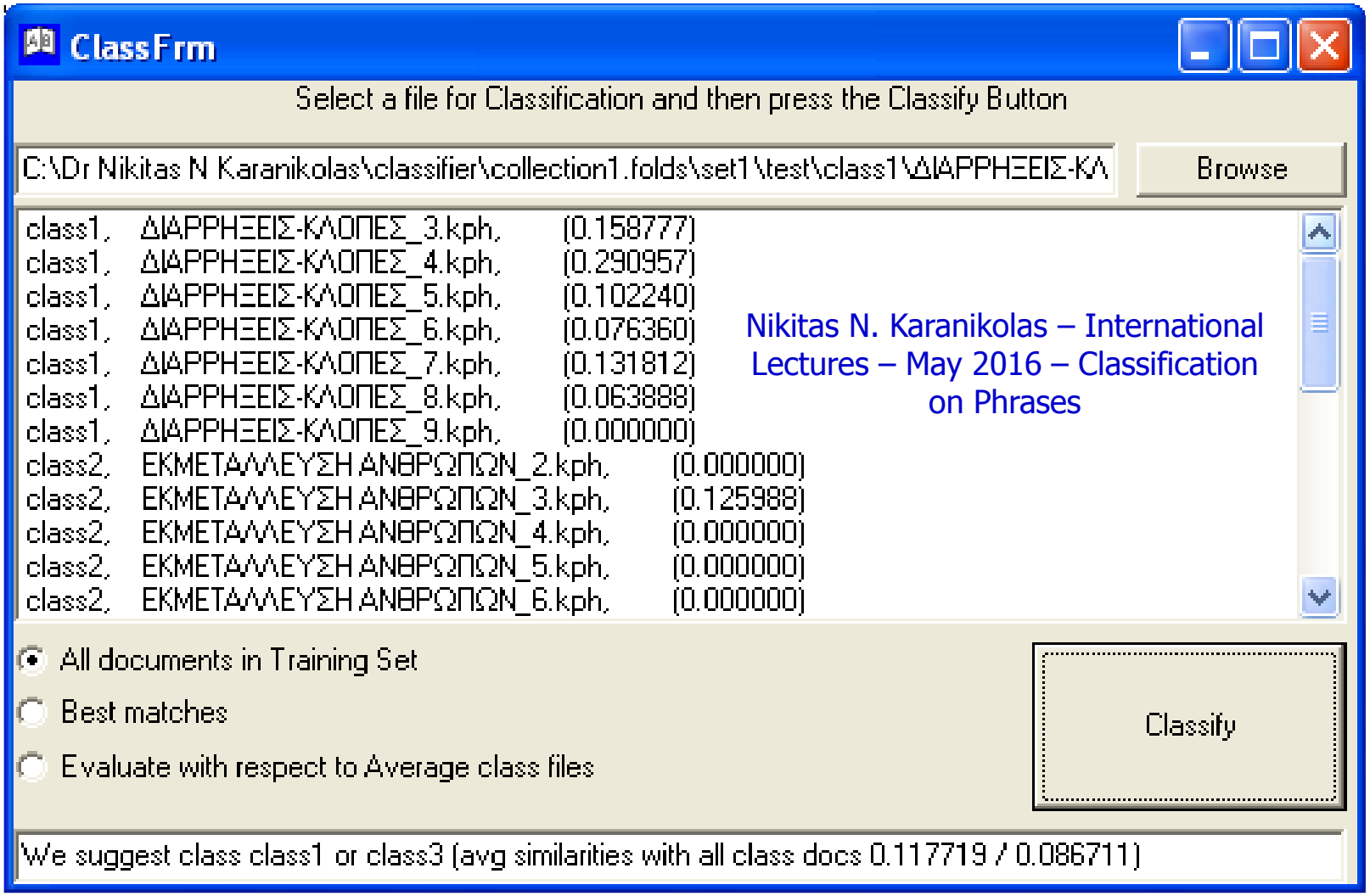
- The parameters of the ALCA algorithm are:
  - the maximum number of phrase constituents (*mpc*),
  - the minimum percentage of texts of a class that must contain a stem in order to use this stem as a seed for generating candidate word phrases (P_0),
  - couples (W_x, P_x) where W_x is the window width that a *x+1* words sequence must exist in order to be taken for candidate *x+1* words key-phrase and P_x is minimum percentage of texts of the class that must contain a candidate *x+1* words key-phrase in order to be accepted as frequent,
  - maximum percentage of classes that can contain a key-phrase in order to not-rejected (P_t).
- Since we focus on key-phrase no longer than four words, the system parameters are P_0, W_1, P_1, W_2, P_2, W_3, P_3 and P_t.
- Only the parameter P_t is provided to the users.
- The rest parameters are controlled through configuration files.
- The decision to provide a user friendly configuration only for the P_t parameter is based on the following estimation: It is better to relax the rest of the constraints for the acceptance of a word-sequence as a key-phrase. Thus a common configuration for the parameters P_0, W_1, P_1, W_2, P_2, W_3, P_3 (that relaxes the constraints for the acceptance of a word-sequence as a key-phrase) is used.

Implementation – The Trainer

- The classification system is based on a couple of programs: The *Trainer* and the *Classifier*.
- The Trainer program conducts two main tasks that the user should follow:
  - creation of the Authority List and also the Class specific Authority sub-Lists,
  - indexing of the training set, which can be based on either the key-phrases of the (Global) Authority List or the Class specific Authority sub-Lists.
- The maximum percentage of classes, that can have the same frequent key-phrase, is determined by the P_t parameter. In the implementation of the Trainer program the user is provided with a handler that adjusts the maximum absolute number of classes (*PtAbsolute*), that can have the same frequent key-phrase.
- When the user activates the first task of Trainer, the ALCA algorithm is activated.
- When the user activates the second task of the trainer, every training document is indexed and in parallel the *Average class documents* are created.

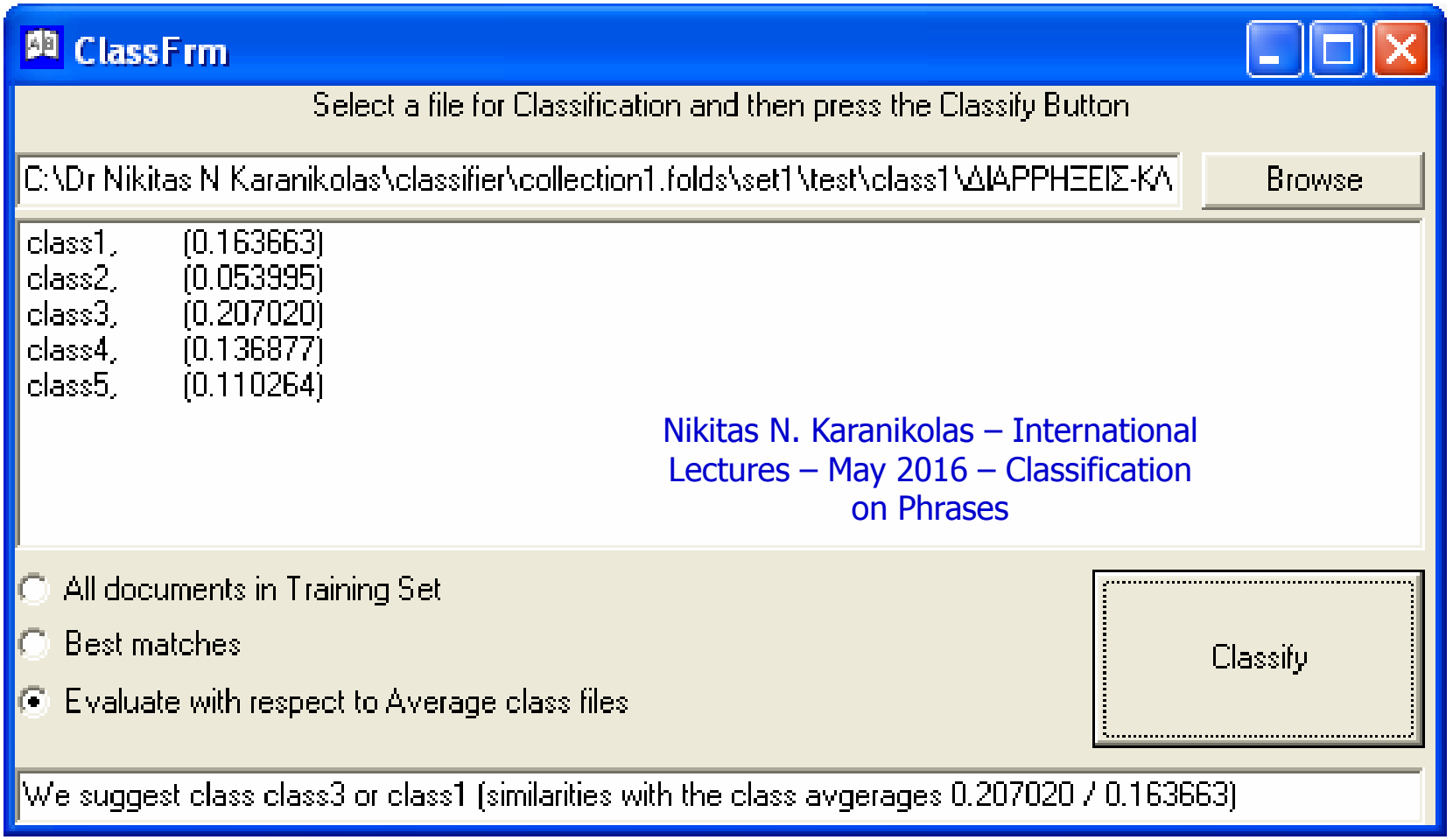
The Trainer's interface

The Classifier's interface



The user defines a text file that contains the new unclassified document (Browse button). Then, selects one of the classification methods: "All documents in Training set" or "Evaluate with respect to Average class files". The answer is given (in the lower dialog item) as a couple of suggestions for the best matching class and the alternative (secondary) matching class.

The Classifier's interface



- The "Evaluate with respect to Average class files" classification method is enabled in the above screenshot and the similarities of the new document with the "Average class documents" are computed.
- The operation of the third (middle) radio button is a variation of the first one ("All documents in the Training Set").

Data – preprocessing - Stemming

- The inflection of the Greek nouns is related to
  - gender (masculine, feminine and neuter),
  - number (singular and plural),
  - case (nominative, genitive, accusative and dative).
- The inflected forms of nouns are usually reflected by suffixes. Therefore, we have a great number of variations for a single noun. This is also the case for Greek verbs.
- The use of Greek verbs follows mood, tenses, voices (active and passive) and inclinations and a single verb has a great number of variations to support the grammatical rules.
- The variation of the inflected forms of nouns and verbs make obvious the need for a lemmatisation lexicon or for some stemming algorithm that replaces words by stems.
- A stemming algorithm has been implemented and has been tested for many years. It follows a similar approach as Porter's algorithm (stepwise replacement of suffixes).
- The application of the stemming algorithm over the documents of the training set is an intermediate step (step 3) of the ALCA algorithm

Stemming & stopword removal example

Title: ΕΙΧΑΝ ΚΑΝΕΙ ΤΗΝ ΕΛΛΑΔΑ ΑΝΩ-ΚΑΤΩ

First paragraph: Από αστυνομικούς της Υ.Δ.Ε.Ζ.Ι. της Υποδιεύθυνσης Ασφαλείας ΝΑ Αττικής και των οικείων Τμημάτων Ασφαλείας, σε συνεργασία με συνοριακούς φύλακες και αστυνομικούς Ο.Π.Κ.Ε. της Α.Δ. Τρικάλων, μετά από αξιοποίηση στοιχείων συνελήφθησαν στα Τρίκαλα και τον Πειραιά 8 αλλοδαποί (7 άνδρες και 1 γυναίκα), ενώ αναζητούνται 5 ακόμη συνεργοί τους που διέφυγαν τη σύλληψη, οι οποίοι αποτελούσαν μέλη 2 ομάδων (σπείρες) που διέπρατταν διακεκριμένες κλοπές και κλοπές πολυτελών Ι.Χ.Ε. αυτοκινήτων.

ΕΛΛ	ΣΥΝ	ΑΝΔΡ	ΣΠΕΙΡ
ΑΝΩ	ΦΥΛ	ΓΥΝΑΙΚ	ΔΙΕΠΡΑΤ
ΑΣΤΥΝΟΜ	ΑΣΤΥΝΟΜ	ΑΝΑΖ	ΔΙΑΚΕΚΡΙ
ΥΠΟΔΙΕΥΘΥΝΣ	ΤΡΙΚΑΛ	ΣΥΝΕΡΓ	ΚΛΟΠ
ΑΣΦΑΛ	ΑΞΙΟΠΟΙ	ΔΙΕΦΥ	ΚΛΟΠ
ΑΤΤ	ΣΤΟΙΧ	ΣΥΛΛΗΨ	ΠΟΛΥΤΕΛ
ΟΙΚ	ΣΥΝΕΛΗΦΘ	ΟΠΟ	ΑΥΤΟΚΙΝ
ΤΜΗ	ΤΡΙΚΑΛ	ΑΠΟΤΕΛ	
ΑΣΦΑΛ	ΠΕΙΡ	ΜΕΛ	
ΣΥΝΕΡΓ	ΑΛΛΟΔΑΠ	ΟΜΑ	

Indexed document with key-phrases

The indexed form based on all the seven paragraphs (as result of the second task of the Trainer) is the following:

000.EURO.	ΔΙΑΡΡΗΞ.ΟΠΟΙ.	ΠΕΡΙΟΧ.ΑΤΤ.
ΑΓΝΩΣΤ.ΑΤΟΜ.	ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.	ΠΟΡΤ.ΠΑΡΑΘΥΡ.
ΑΡΜΟΔ.ΕΙΣΑΓΓΕΛ.	ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.ΚΛΟΠ.	ΤΙΜΑΛΦ.ΗΛΕΚΤΡ.
ΑΣΦΑΛ.ΑΤΤ.	ΔΙΕΠΡΑΤ.ΚΛΟΠ.	ΤΡΟΠ.ΔΡΑ.
ΑΣΦΑΛ.ΣΥΝΕΡΓ.	ΔΡΑ.ΣΥΓΚΕΚΡΙ.	ΤΡΟΠ.ΔΡΑ.ΣΥΓΚΕΚΡΙ.
ΑΥΤΟΚΙΝ.ΟΠΟΙ.	ΚΑΤΟΧ.ΚΑΤΑΣΧΕΘ.	ΤΡΟΠ.ΣΥΓΚΕΚΡΙ.
ΑΦΑΙΡ.ΑΥΤΟΚΙΝ.	ΚΑΤΟΧ.ΚΛΕ.	ΦΟΡ.ΔΙΕΠΡΑΤ.
ΑΦΑΙΡ.ΤΙΜΑΛΦ.	ΚΛΕ.ΑΥΤΟΚΙΝ.	ΦΟΡ.ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.
ΑΦΑΙΡ.ΤΙΜΑΛΦ.ΗΛΕΚΤΡ.	ΚΛΕ.ΑΥΤΟΚΙΝ.ΟΠΟΙ.	ΦΟΡ.ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.ΚΛΟΠ.
ΑΦΑΙΡ.ΧΡΗ.	ΚΛΟΠ.ΔΙΑΡΡΗΞ.	ΦΟΡ.ΔΙΕΠΡΑΤ.ΚΛΟΠ.
ΑΦΑΙΡ.ΧΡΗ.ΗΛΕΚΤΡ.	ΜΕΛ.ΣΠΕΙΡ.	ΧΑΡΑΚΤΗΡ.ΔΡΑ.
ΑΦΑΙΡ.ΧΡΗ.ΤΙΜΑΛΦ.	ΟΔΗΓ.ΑΡΜΟΔ.ΕΙΣΑΓΓΕΛ.	ΧΑΡΑΚΤΗΡ.ΔΡΑ.ΣΥΓΚΕΚΡΙ.
ΒΑΡ.ΟΔΗΓ.	ΟΔΗΓ.ΕΙΣΑΓΓΕΛ.	ΧΑΡΑΚΤΗΡ.ΤΡΟΠ.
ΒΑΡ.ΟΔΗΓ.ΕΙΣΑΓΓΕΛ.	ΟΜΑ.ΦΟΡ.	ΧΑΡΑΚΤΗΡ.ΤΡΟΠ.ΔΡΑ.ΣΥΓΚΕΚΡΙ.
ΔΙΑΚΕΚΡΙ.ΚΛΟΠ.	ΟΜΑ.ΦΟΡ.ΔΙΕΠΡΑΤ.	ΧΑΡΑΚΤΗΡ.ΤΡΟΠ.ΣΥΓΚΕΚΡΙ.
ΔΙΑΜ.ΠΕΡΙΟΧ.	ΟΜΑ.ΦΟΡ.ΔΙΕΠΡΑΤ.ΔΙΑΚΕΚΡΙ.	ΧΡΗ.ΗΛΕΚΤΡ.
ΔΙΑΜ.ΠΕΡΙΟΧ.ΑΤΤ.	ΟΜΑ.ΦΟΡ.ΔΙΕΠΡΑΤ.ΚΛΟΠ.	ΧΡΗ.ΤΙΜΑΛΦ.

Data – Training sets – 1st collection

- We used two different collections of documents
- The first collection (from the magazine “Police Inspection” and the issues of 2004 and 2005) has 5 classes
  - housebreakings and robberies (in Greek: “διαρρήξεις και κλοπές”) items 9
  - pimping, pandering and presumes upon humans (“μαστροπεία και εκμετάλλευση ανθρώπων”) items 6
  - electronic crime (“ηλεκτρονικό έγκλημα”) items 4
  - drugs (“ναρκωτικά”) items 11
  - forgery (“πλαστογραφία) items 5

Data – Training sets – 2nd collection

- The second collection has 7 classes
- Documents are patient discharge letters from the Areteion University Hospital (Athens, Greece):
  - “obstruent icterus” (ICD9 code: 0010, Greek term: “αποφρακτικός ίκτερος”) 4 items
  - “Echinococcosis, unspecified, of liver” (122.8, “Εχινοκοκκίαση του ήπατος, μη καθορισμένη”) 4 items
  - “Malignant neoplasm of stomach” (151, “κακοήθη νεοπλάσματα του στομάχου”) 4 items
  - “Malignant neoplasm of colon” (153, “κακοήθη νεοπλάσματα παχέος εντέρου, πλήν ορθού”) 4 items
  - “Malignant neoplasm of Sigmoid” (153.3, “κακοήθη νεοπλάσματα σιγμοειδούς”) 5 items
  - “Malignant neoplasm of rectum” (154.1, “κακοήθη νεοπλάσματα ορθού”) 4 items
  - “Malignant neoplasm of liver, primary” (155.0, “κακοήθη νεοπλάσματα ήπατος, πρωτοπαθές”). 4 items

Document from the 2nd collection

Discharge Diagnosis
Primary malignant neoplasm of the liver
Admitting Diagnosis
Liver cancer
Past history and Presentation
71 years old male patient with a history of a AAA (Abdominal Aortic Aneurysm) repair 13 months ago suffers with RUQ (Right Upper Quadrum) pain and a palpable mass of two months duration. An abdominal CT scan showed a 12 cm tumor in the right lobe of the liver and a smaller one in segment V. He is admitted for surgical treatment.
Progress notes
The pt. underwent a full pre-op evaluation including vascular consultation and cardiac and pulmonary evaluation. An MRI showed a multifocal liver tumor that proved to be by a FNA (Fine Needle Aspiration) a moderately differentiated primary hepatocellular carcinoma. An upper and lower GI (Gastro Intenstinal) endoscopy, as well as a chest CT scan were negative. On 31/8/00 he underwent an extended right hepatectomy and was transferred to the ICU (Intensive Care Unit). His post-op course was complicated by pulmonary and liver failure. His bilirubin reached 12.7mg/dl and his AST, ALT and γGT were markedly increased. His renal function also deteriorated with a peak creatinine level of 2.9 mg/dl. Eventually his condition improved and was transferred to the ward where he was treated with diuretics and TPN (Total Parenteral Nutrition). On POD (Post Operative Date) #8 he developed wound infection and the wound was opened and drained. He also received appropriate antibiotics. He developed also decubitus ulcers, which were surgically debrided. On POD#15 his condition improved, was started on oral feedings and was ambulated. His albs showed improvement. Eventually his ascites diminished and his liver and renal function improved. He was discharged with the following labs: bil=4.9 mg/dl, alb=2.3g/dl, cr=1.75 mg/dl, γGT=136, WBC=6600 and Ht=35.3%
Discharge orders
High protein diet
Return for a follow up visit in 4 months

Experiments – data folds

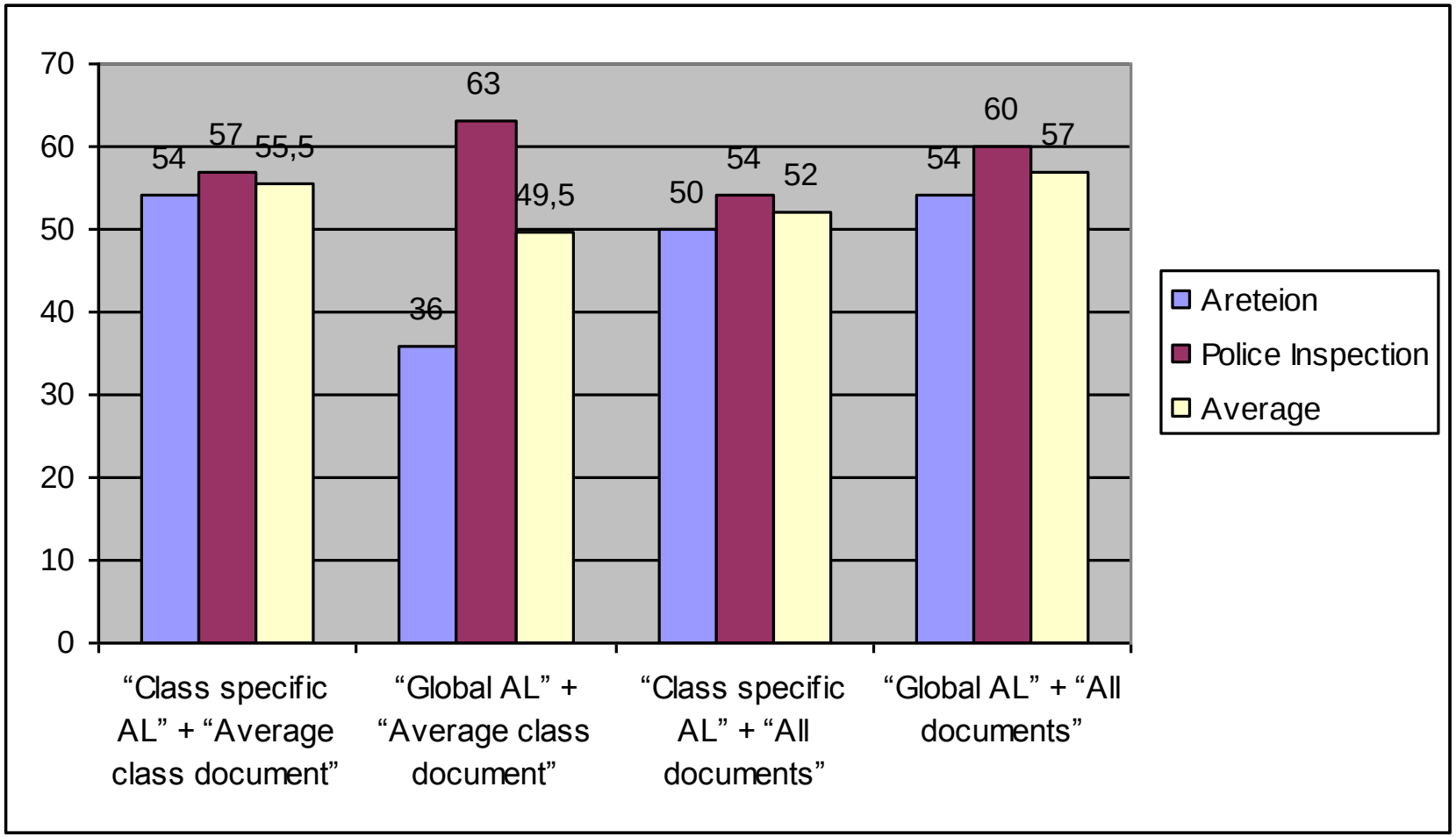
	Training Set documents (TS) for the Class1	TS for the Class2	TS for the Class3	TS for the Class4	TS for the Class5
Fold1	3 .. 9	2 .. 6	2 .. 4	3 .. 11	2 .. 5
Fold2	1, 2, 5 .. 9	1, 3 .. 6	1, 3, 4	1, 2, 5 .. 11	1, 3 .. 5
Fold3	1 .. 4, 7 .. 9	1, 2, 4 .. 6	1, 2, 4	1 .. 4, 7 .. 11	1, 2, 4, 5
Fold4	1 .. 6, 9	1 .. 3, 5, 6	1 .. 3	1 .. 6, 9 .. 11	1 .. 3, 5
Fold5	1 .. 8	1 .. 4	1 .. 4	1 .. 8	1 .. 4

4 folds in a similar way defined for the 2nd collection

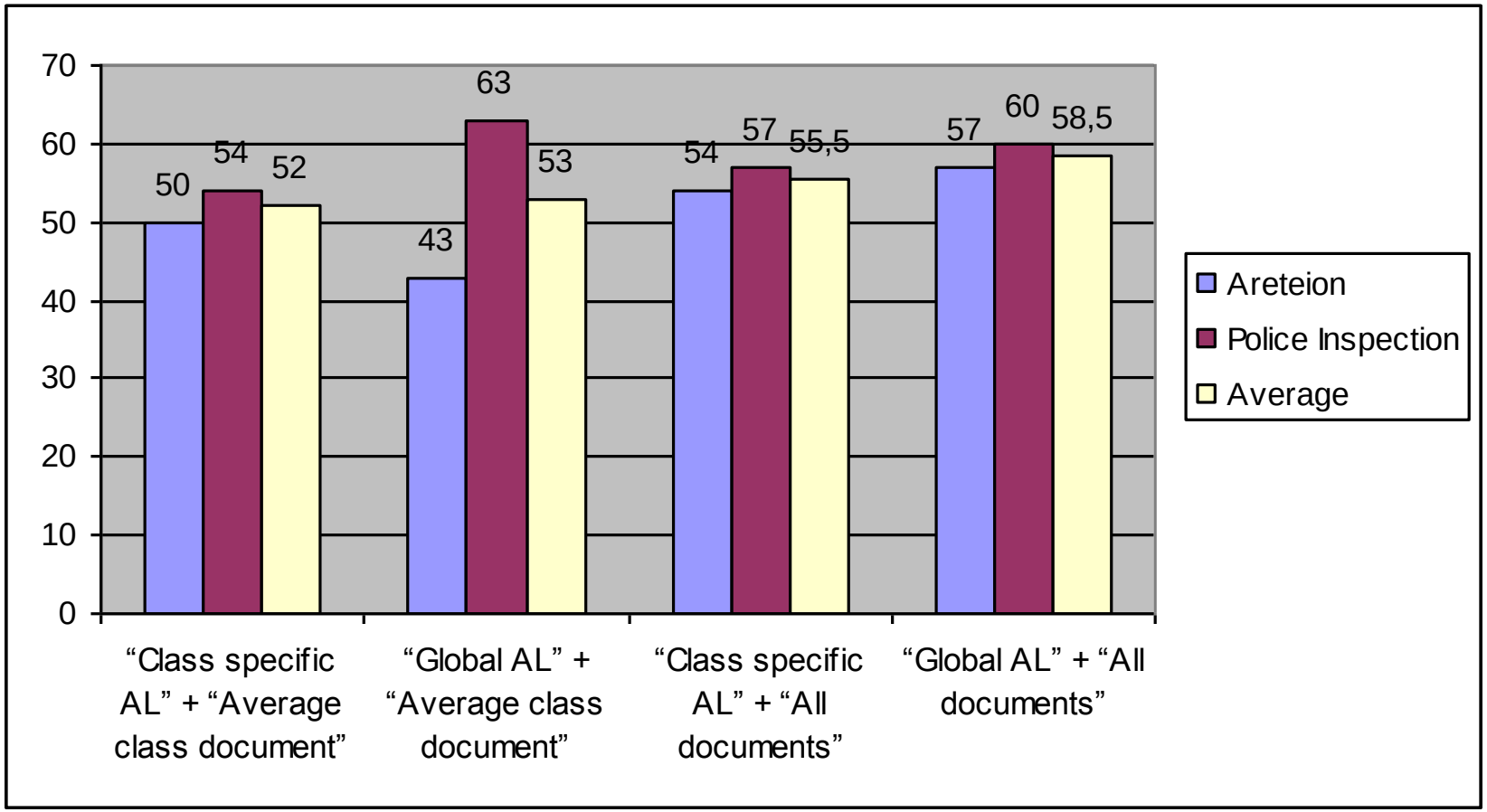
Expriments

- We executed the Trainer program and chose the use of (Global) Authority List. Next, we executed the Classifier program and chose the Average class document (*S'*), in order to classify the test documents of the elaborated combination. We counted how many documents were correctly classified and how many documents were erroneously classified.
- We repeated the classification of test documents using All documents (in Training Set) (*S* and *S''*) instead of using Average class document. Again, we counted the correctly classified documents and the erroneously classified ones.
- We executed the Trainer program but now chose the Class specific Authority sub-Lists instead of (Global) Authority List. We executed the Classifier using, consecutively, Average class document (*S'*) and All documents (*S* and *S''*).

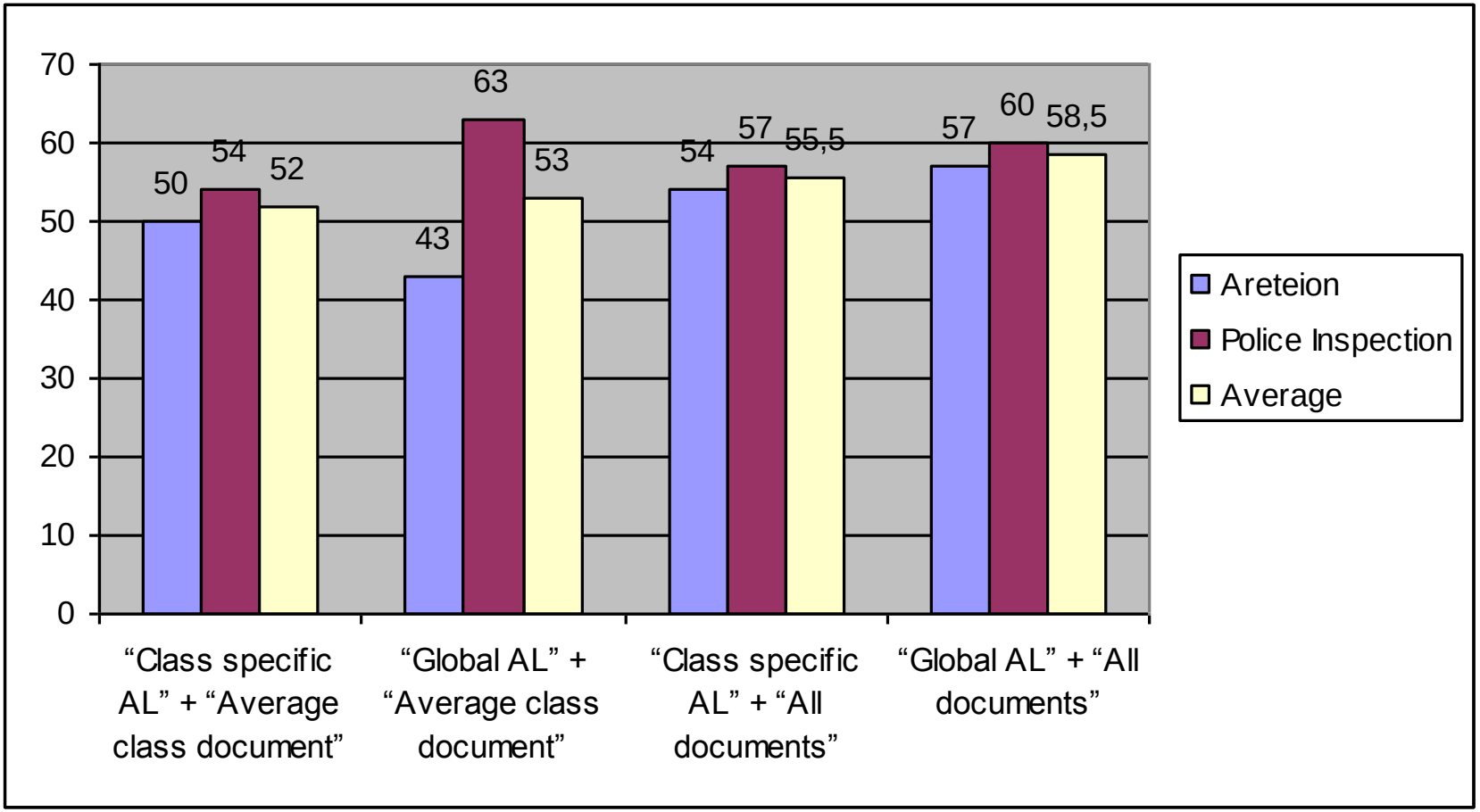
Results according to the first systems' suggestion with PtAbsolute=1



Results according to the first systems' suggestion with PtAbsolute=2



Results according to the first systems' suggestion with PtAbsolute=3



Conclusions (1/2)

- According to the results presented and considering the average collection results (yellow – third – bar in each triplet of bars, in each figure), we can conclude that the classification method “All documents” performs better than the “Average class document” classification method.
- Moreover, by the same figures, we can conclude that the usage of “(Global) Authority List” (“Global AL”) performs marginally better than the usage of “Class specific AL”.
- The best combination is the classification method **“All documents”** (activated by the Classifier’s interface) with the creation and usage of the **“Global AL”** (activated by the Trainer’s interface).

Conclusions (2/2)

- The average success rate of the system when the best systems parameters are used (“All documents”, “Global AL” and *PtAbsolute* in {2|3}) is **58,5%** whenever only the first systems suggestion is used.
- It is increased to **75,0%** when both systems suggestions are used.
- the “Global AL” does not perform definitely better than the “Class specific AL” and should be evaluated for each specific collection in conjunction with the other parameters
- The results given with the “Average class document” method are not so disappointing. This gives us permittion to build cheap text classification systems using the “Average class document” method.

Based on

- **N. Karanikolas**, C. Skourlas, A. Christopoulou and T. Alevizos. Medical Text Classification based on Text Retrieval techniques. MEDINF 2003. 1st International Conference on Medical Informatics & Engineering, October 9 - 11, 2003, Craiova, Romania. Craiova Medicala Journal, volume 5, supplement 3, ISSN 1454-6876.
- **N. Karanikolas** and C. Skourlas. Key-Phrase Extraction for Classification. MEDICON and HEALTH TELEMATICS 2004. X Mediterranean Conference on Medical and Biological Engineering, 31 July - 5 August, 2004, Ischia, Italy. In IFMBE Proceedings, Health in the Information Society, volume 6, ISBN 88-7080308-8, ISSN 1727-1983.
- **Nikitas Karanikolas** and C. Skourlas. Text Classification: Forming Candidate Key-Phrases from Existing Shorter Ones. FACTA UNIVERSITATIS Series: Electronics and Energetics, ISSN 0353-3670, volume 19, No 3, 2006.
- **N. Karanikolas** and C. Skourlas. A parametric methodology for text classification. Journal of Information Science, Vol. 36 (4), pp. 421-442, 2010, doi:10.1177/0165551510368620.

Text classification based on phrases

- Thank you for your attention,
- I will try to answer Questions.

Text classification based on phrases

- **AUTHOR:**
- Nikitas N. Karanikolas,
- Professor,
- Dept. of Informatics,
- Technological Educational Institute (TEI) of Athens,
- http://users.teiath.gr/nnk/
- nnk@teiath.gr