TRECVID-2006 High-Level Feature task: Overview

Wessel Kraaij
TNO
&
Paul Over
NIST

Outline

- Task summary
- Evaluation details
  - Inferred Average precision vs. mean average precision
  - Participants
- Evaluation results
  - Pool analysis
  - Results per category
  - Results per feature
  - Significance tests category A
  - comparison with TV2005
- Global Observations
- Issues

High-level feature task

- Goal: Build benchmark collection for visual concept detection methods
- Secondary goals:
  - encourage <u>generic</u> (scalable) methods for detector development
  - feature-indexing could help search/browsing
- Participants submitted runs for all 39 LSCOM-lite features
- Used results of 2005 collaborative training data annotation
  - Tools from CMU and IBM (new tool)
  - 39 features and about 100 annotators
  - multiple annotations of each feature for a given shot
- Range of frequencies in the common development data annotation
- NIST evaluated 20 (medium frequency) features from the 39 using a 50% random sample of the submission pools (Inferred AP)

HLF is challenging for machine learning

- Small imbalanced training collection
- Large variation in examples
- Noisy Annotations

- Decisions to be made:
  - find suitable representations
  - find optimal fusion strategies

20 LSCOM-lite features evaluated

1 sports
3 weather
5 office
6 meeting
10 desert
12 mountain
17 waterscape/waterfront
22 corporate leader
23 police security
24 military personnel
26 animal
27 computer tv screen
28 us flag
29 airplane
30 car
32 truck
35 people marching
36 explosion fire
38 maps
39 charts

Note: this is a departure from the numbering scheme used at previous TV's

High-level feature evaluation

- Each feature assumed to be binary: absent or present for each master reference shot
- Task: Find shots that contain a certain feature, rank them according to confidence measure, submit the top 2000
- NIST pooled and judged top results from all submissions
- Evaluated performance effectiveness by calculating the *inferred average precision* of each feature result
- Compared runs in terms of **mean** *inferred average precision* across the 20 feature results
  - to be used for comparison between TV2006 HLF runs
  - not comparable with TV2005, TV2004… figures

Inferred average precision (infAP)

- Just* developed by Emine Yilmaz and Javed A. Aslam at Northeastern University
- Estimates average precision surprisingly well using a surprisingly small sample of judgments from the usual submission pools
- Experiments on TRECVID 2005 feature submissions confirmed quality of the estimate in terms of actual scores and system ranking

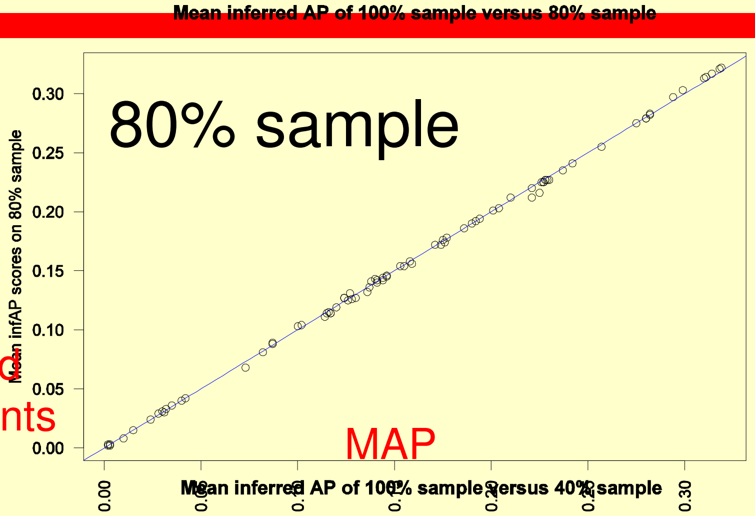
* J.A. Aslam, V. Pavlu and E. Yilmaz, ***Statistical Method for System Evaluation Using Incomplete Judgments*** Proceedings of the 29th ACM SIGIR Conference, Seattle, 2006.

Inferred average precision (infAP) Experiments with 2005 data

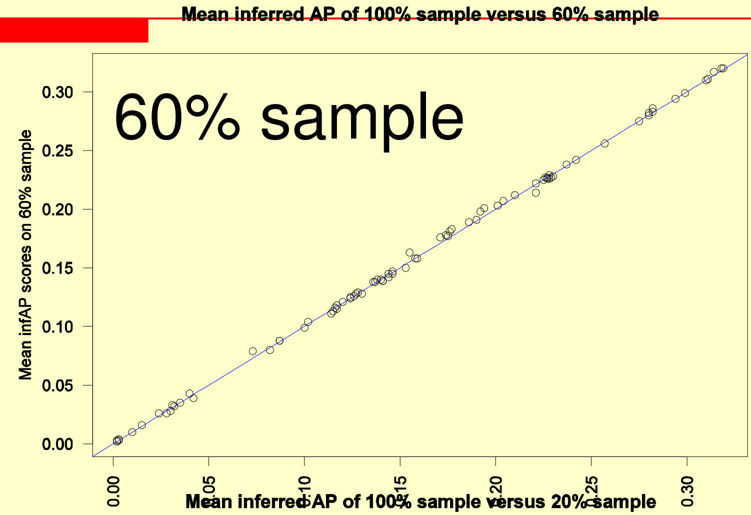
- Pool submitted results down to at least a depth of 200 items
- Manually judge pools - forming a base set of judgments (100% judged)
- Create 4 sampled sets of judgments by randomly marking some results "unjudged"
  - 20% unjudged -> 80% sample
  - 40% unjudged -> 60% sample
  - 60% unjudged -> 40% sample
  - 80% unjudged -> 20% sample
- Evaluate all systems that submitted results for all features in 2005 using the base and each of the 4 sampled judgment sets using infAP
- By definition, infAP of a 100% sample of the base judgment set is identical to average precision (AP).
- Compare measurements of infAP using various sampled judgment sets to standard AP.

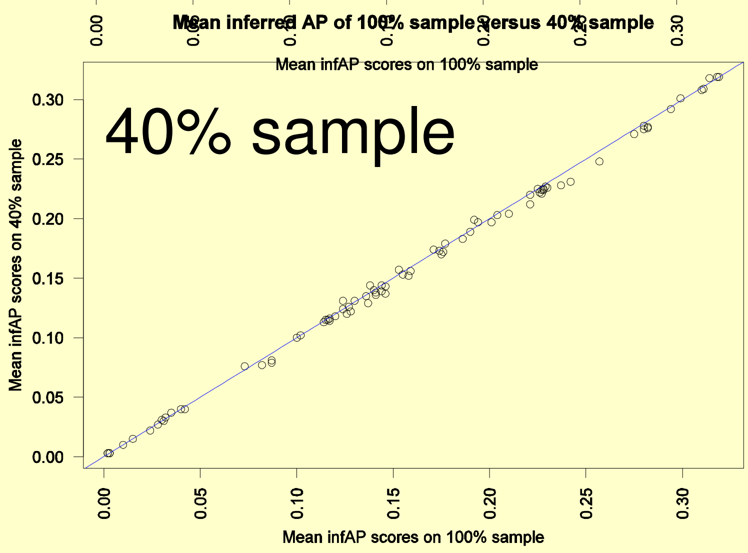
2005 Mean InfAP scoring approximates MAP scoring very closely

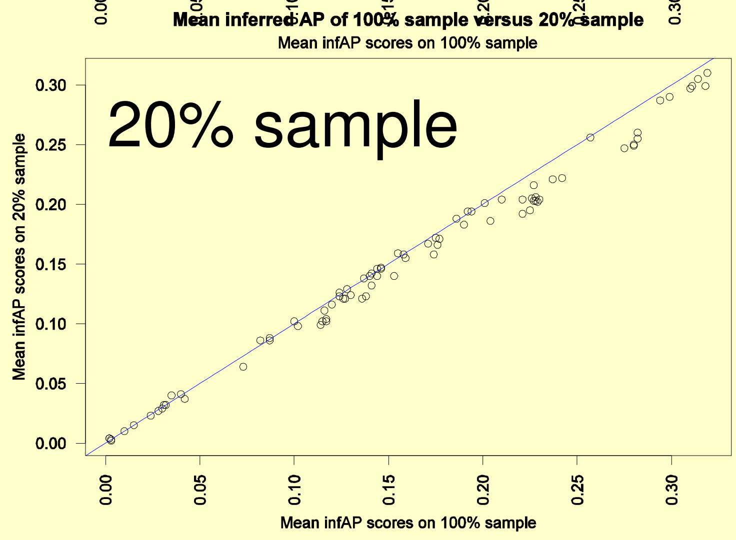
mean
infAP
on
sampled
judgments



MAP







2005 system rankings change very little when determined based on infAP versus AP.

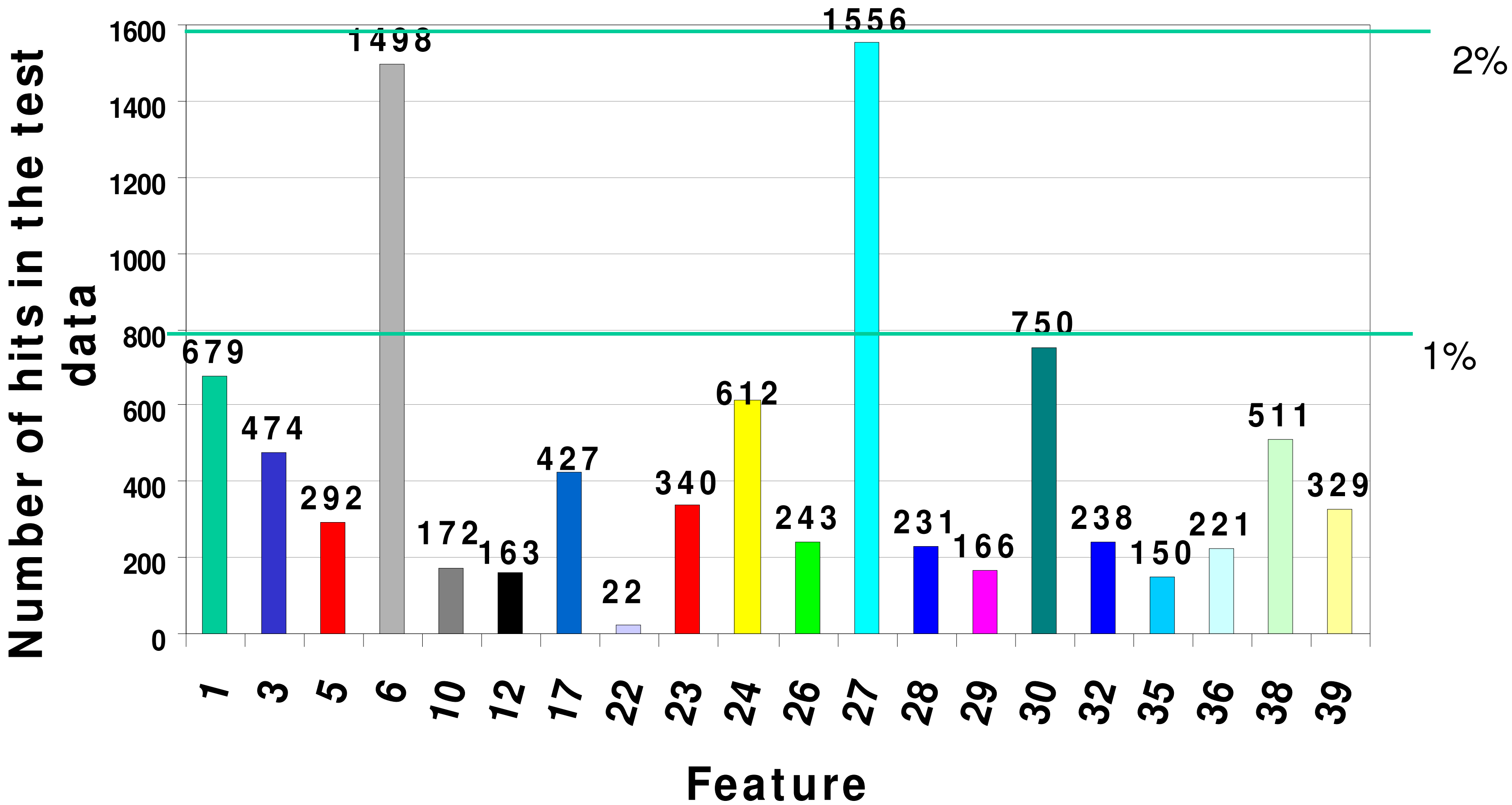
- Kendall's tau (normalizes pairwise swaps)
  - 80% sample 0.9862658
  - 60% sample 0.9871663
  - 40% sample 0.9700546
  - 20% sample 0.951566
- Number of significant rank changes (randomization test, $p<0.01$)

	Swap	Lose	Keep	Add
80%	0	35	2018	37
60%	0	57	1996	36
40%	0	104	1949	45
20%	0	170	1883	73

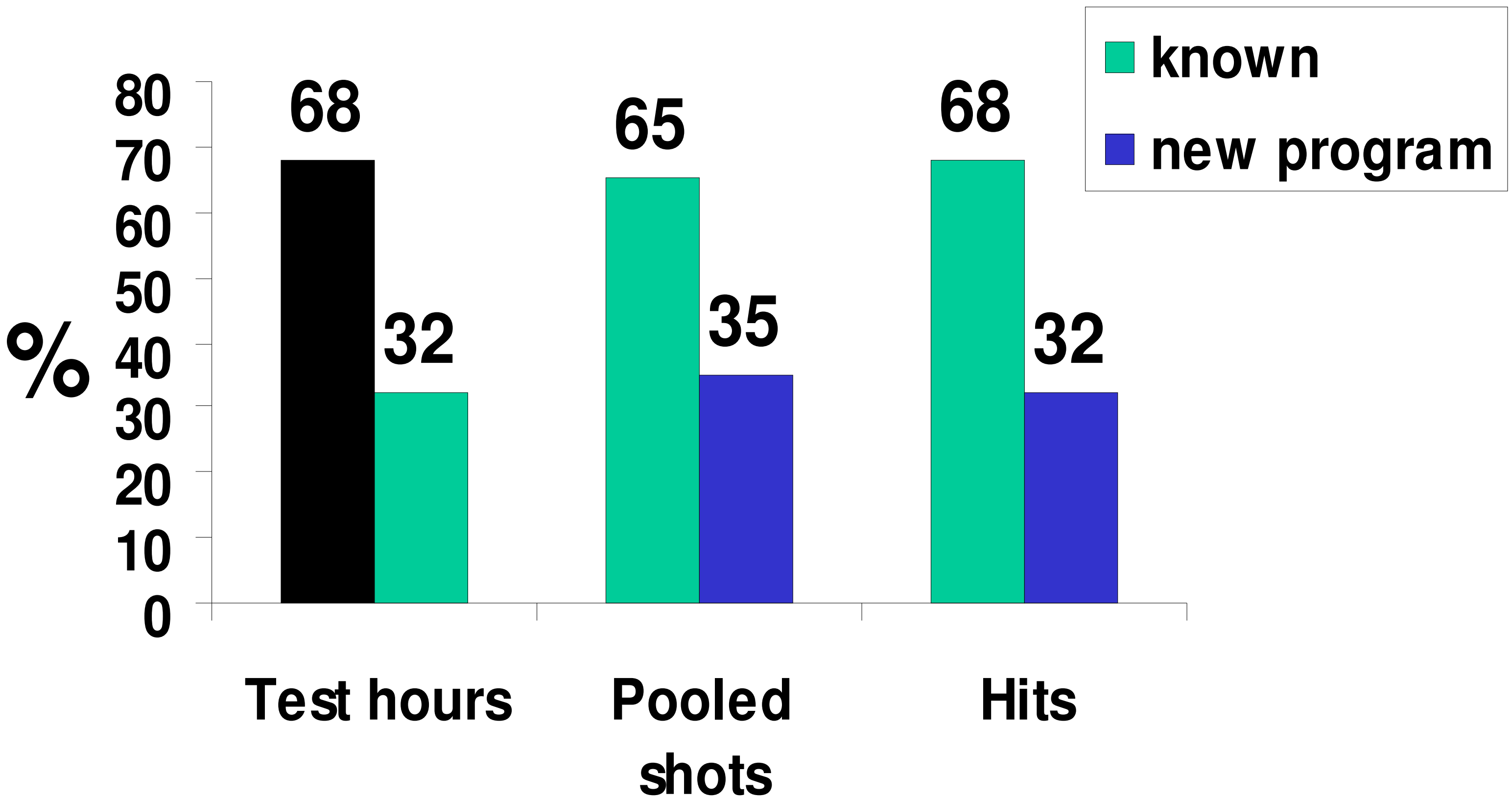
2006: Inferred average precision (infAP)

- Submissions for each of 20 features were pooled down to about 120 items (so that each feature pool contained ~ 6500 shots)
  - varying pool depth per feature
- A 50% random sample of each pool was then judged:
- 66,769 total judgements (~ 125 hr of video)
- Judgement process: one assessor per feature, watched complete shot while listening to the audio.
- infAP was calculated using the judged and unjudged pool by trec_eval

Frequency of hits varies by feature



Systems can find hits in video from programs not in the training data



2006: 30/54 Participants (2005: 22/42, 2004: 12/33)

Bilkent U.	-- FE SE --
Carnegie Mellon U.	-- FE SE --
City University of Hong Kong (CityUHK)	SB FE SE --
CLIPS-IMAG	SB FE SE --
Columbia U.	-- FE SE --
COST292 (www.cost292.org)	SB FE SE RU
Fudan U.	-- FE SE --
FX Palo Alto Laboratory Inc	SB FE SE --
Helsinki U. of Technology	SB FE SE --
IBM T. J. Watson Research Center	-- FE SE RU
Imperial College London / Johns Hopkins U.	-- FE SE --
NUS / I2R	-- FE SE --
Institut EURECOM	-- FE -- RU
KDDI/Tokushima U./Tokyo U. of Technology	SB FE -- --
K-Space (kspace.qmul.net)	-- FE SE --

2006: 30 Participants (continued)

LIP6 - Laboratoire d'Informatique de Paris 6	-- FE -- --
Mediamill / U. of Amsterdam	-- FE SE --
Microsoft Research Asia	-- FE -- --
National Taiwan U.	-- FE -- --
NII/ISM	-- FE -- --
Tokyo Institute of Technology	SB FE -- --
Tsinghua U.	SB FE SE RU
U. of Bremen TZI	-- FE -- --
U. of California at Berkeley	-- FE -- --
U. of Central Florida	-- FE SE --
U. of Electro-Communications	-- FE -- --
U. of Glasgow / U. of Sheffield	-- FE SE --
U. of Iowa	-- FE SE --
U. of Oxford	-- FE SE --
Zhejiang U.	SB FE SE --

HLF keeps attracting more participants, most of them come back the next year.

Number of runs of each training type

Tr-Type	2006	2005	2004	2003
A	86 (68.8%)	79 (71.8%)	45 (54.2%)	22 (36.7%)
B	32 (25.6%)	24 (21.8%)	27 (32.5%)	20 (33.3%)
C	7 (5.6%	7 (6.3%)	11 (13.3%)	18 (30.0%)
Total runs	125	110	83	60

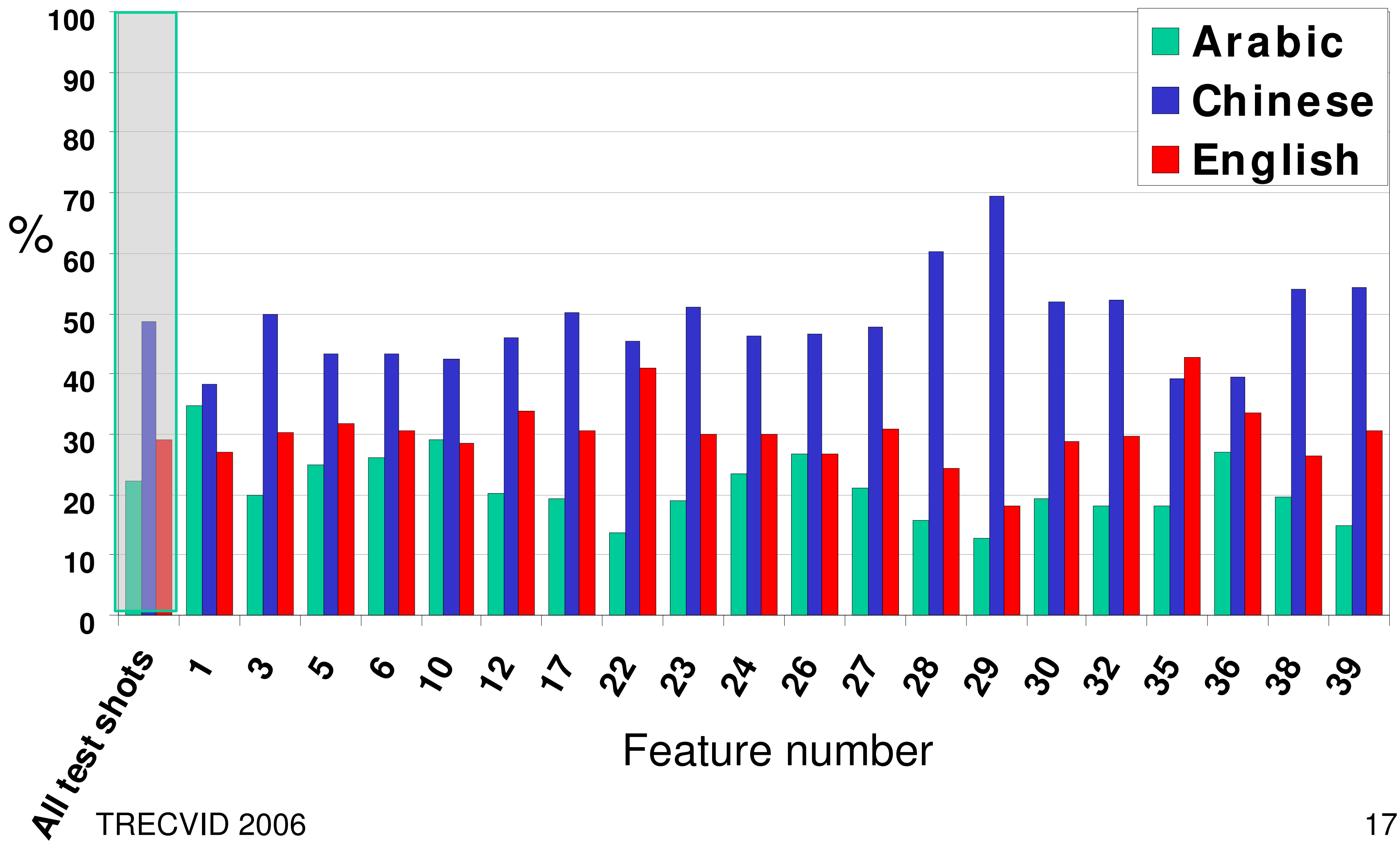
System training type:

A - Only on common dev. collection and the common annotation

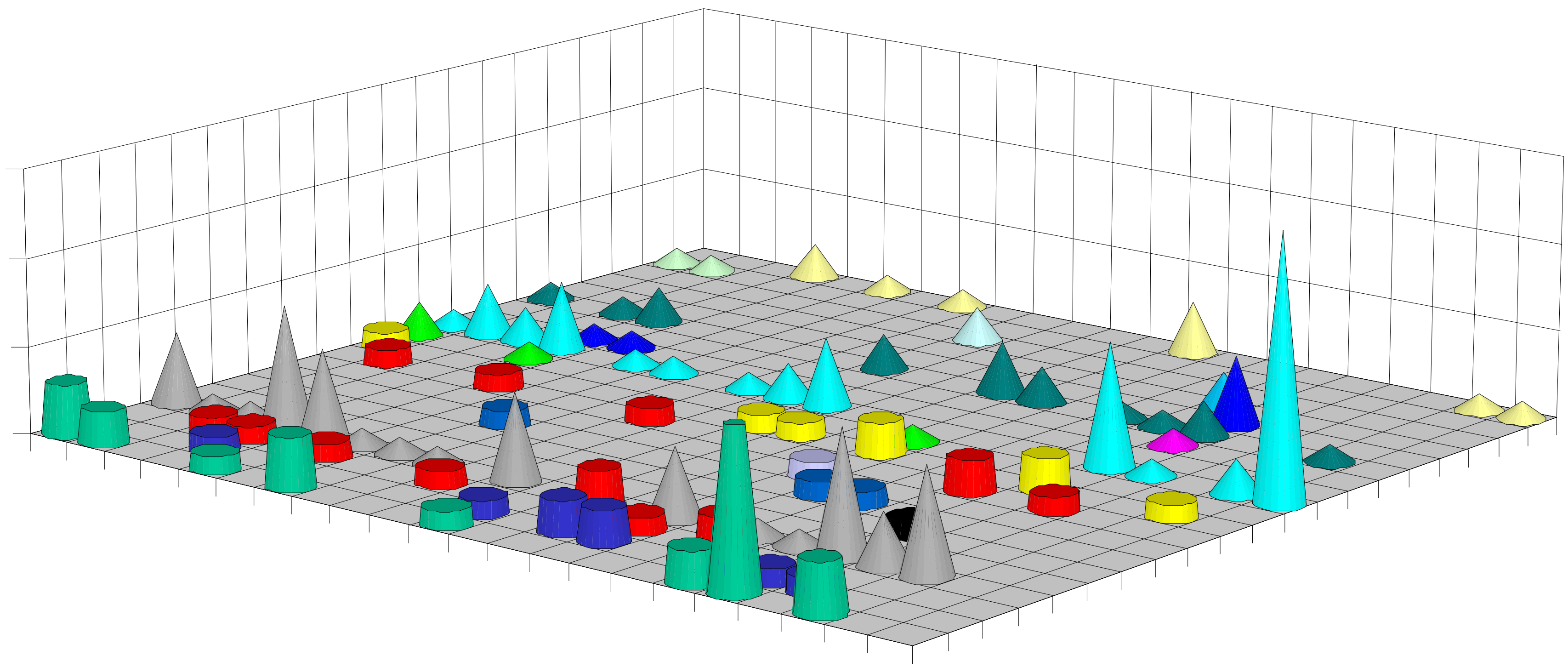
B - Only on common dev. collection but not on (just) the common annotation

C - not of type A or B

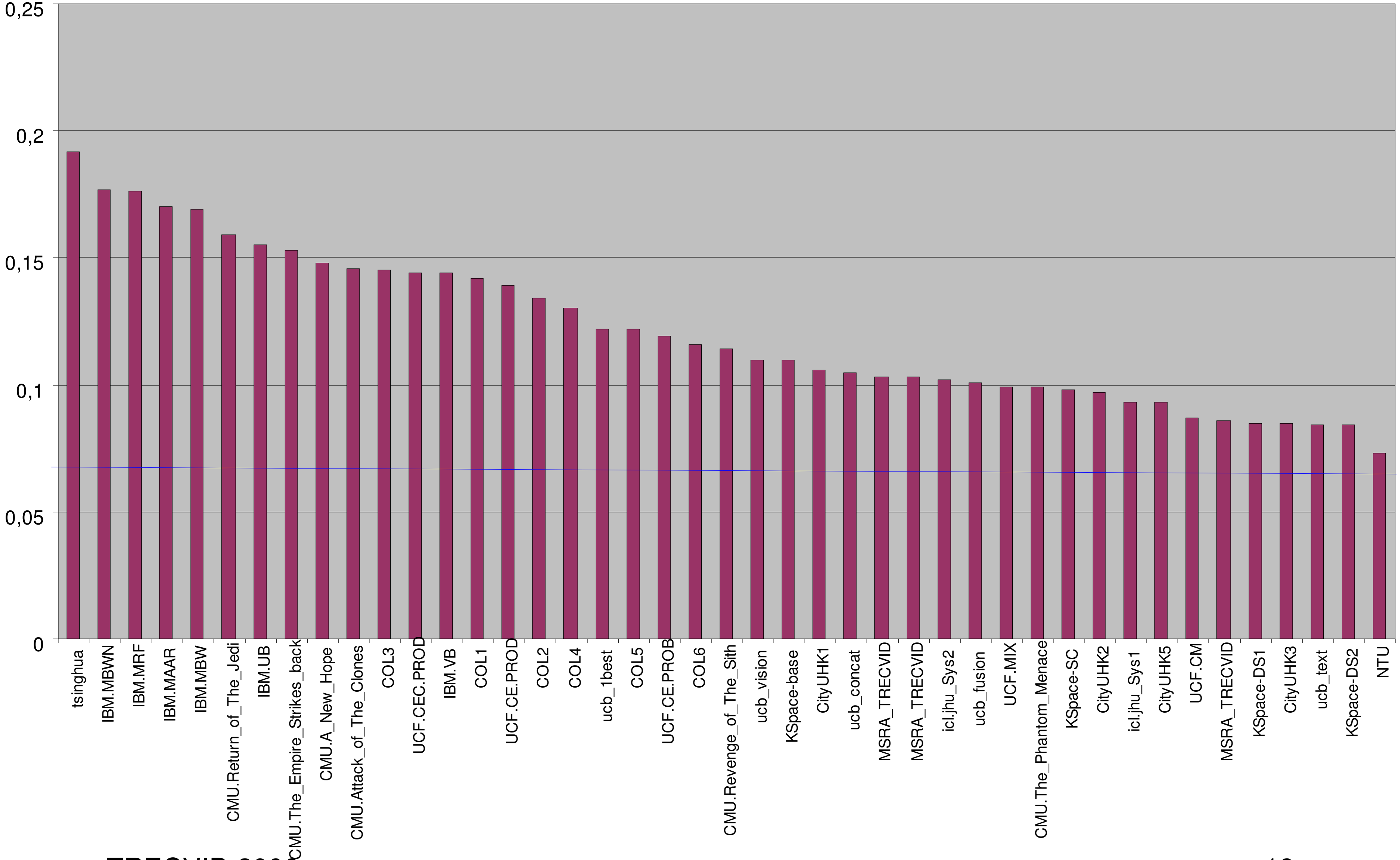
% of true shots by source language for each feature



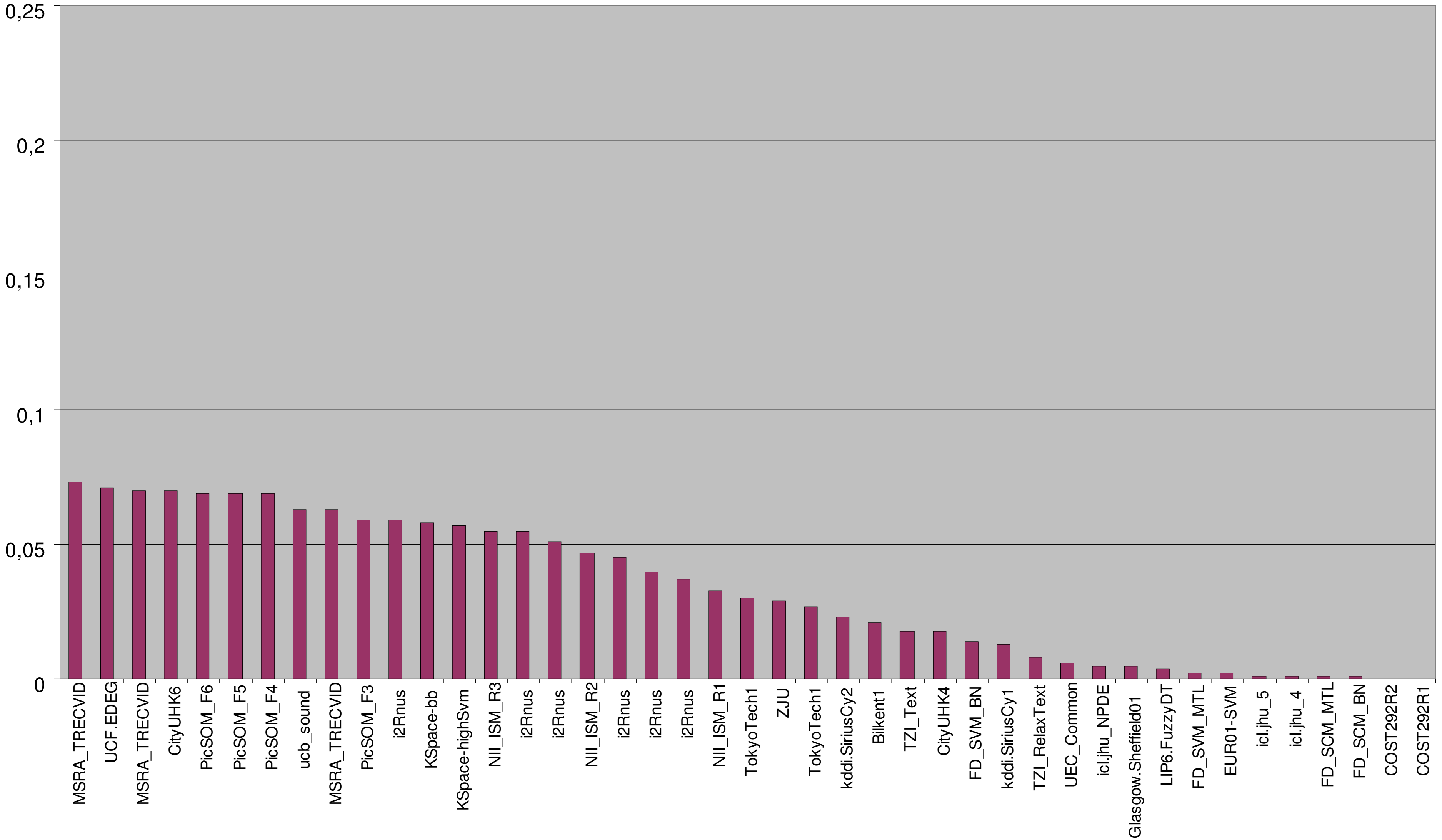
True shots contributed uniquely by team for each feature



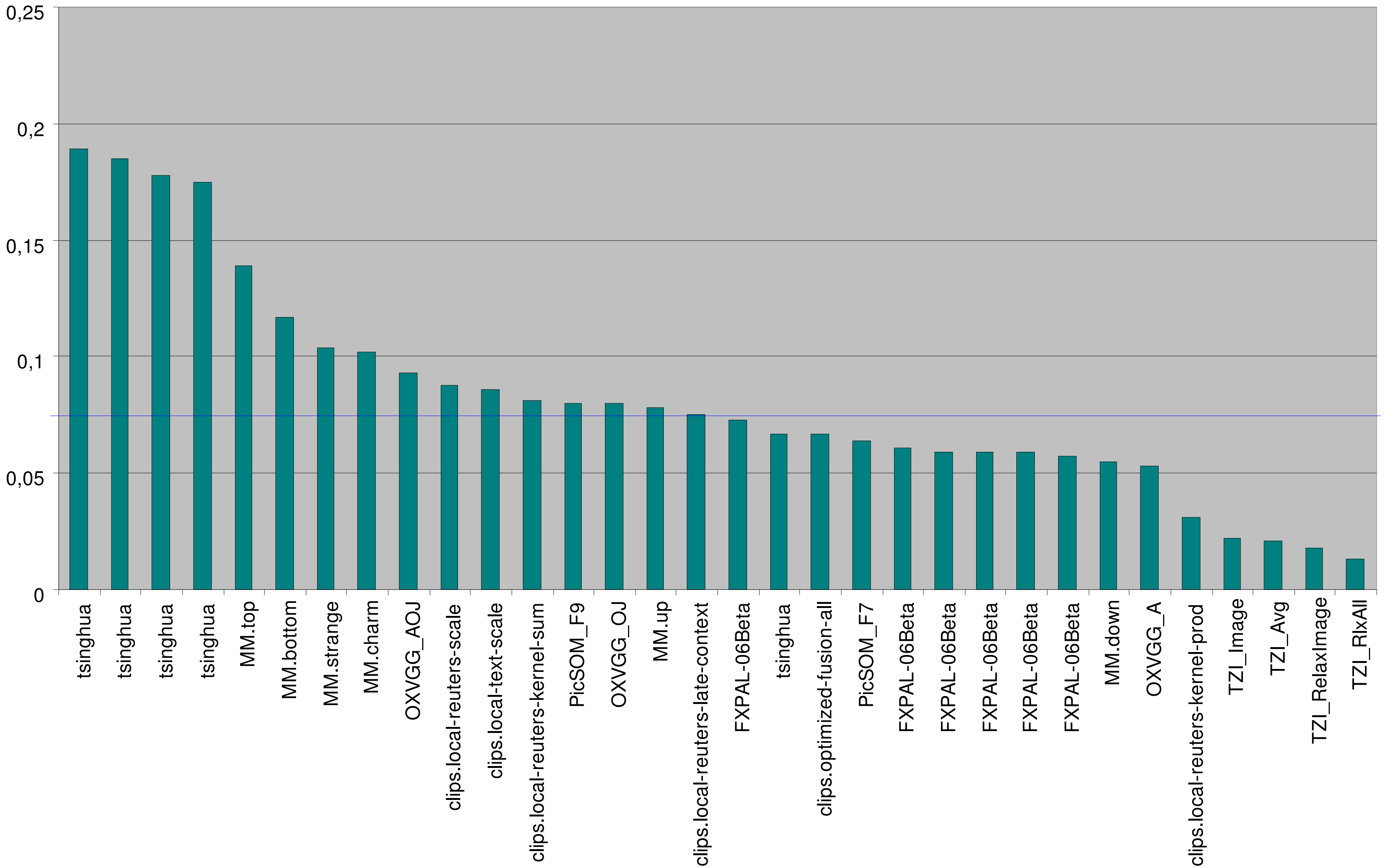
Category A results (top half)



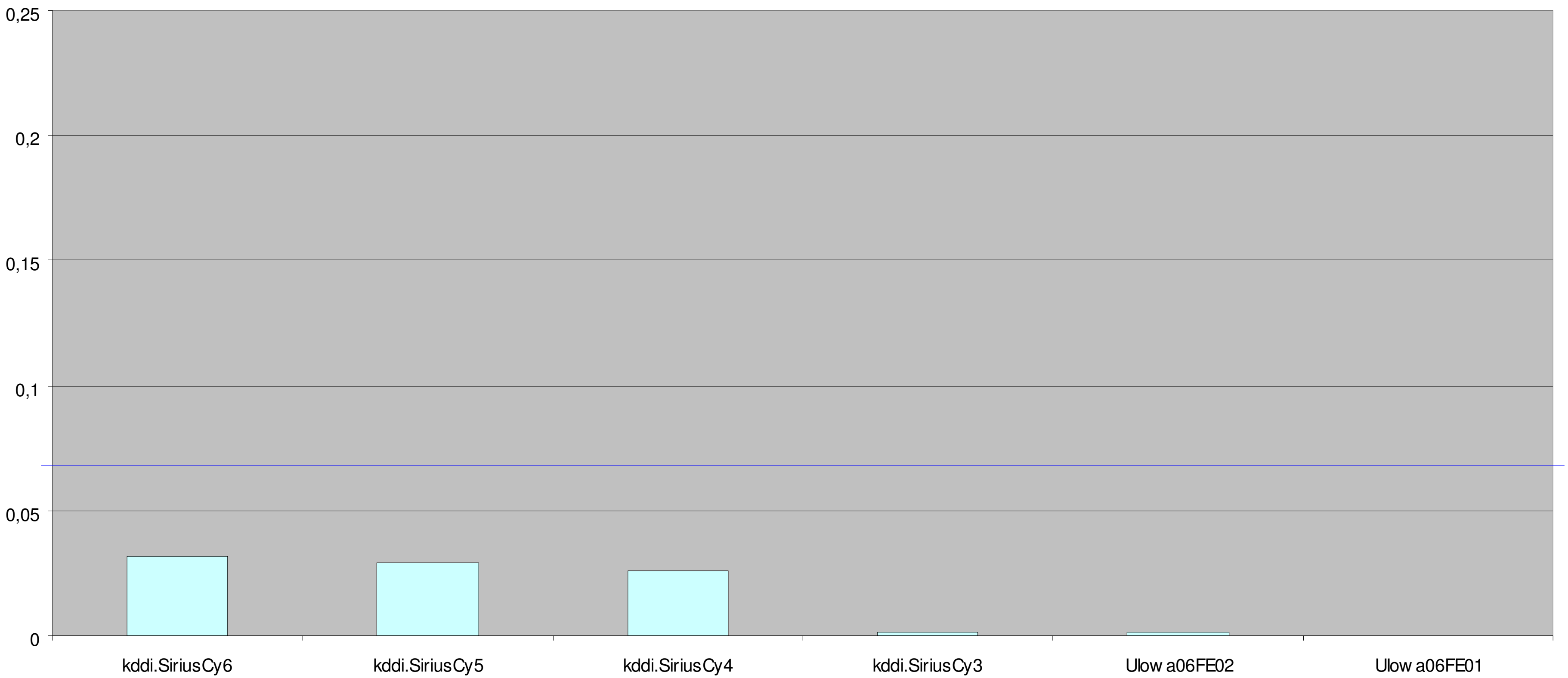
Category A (bottom half)



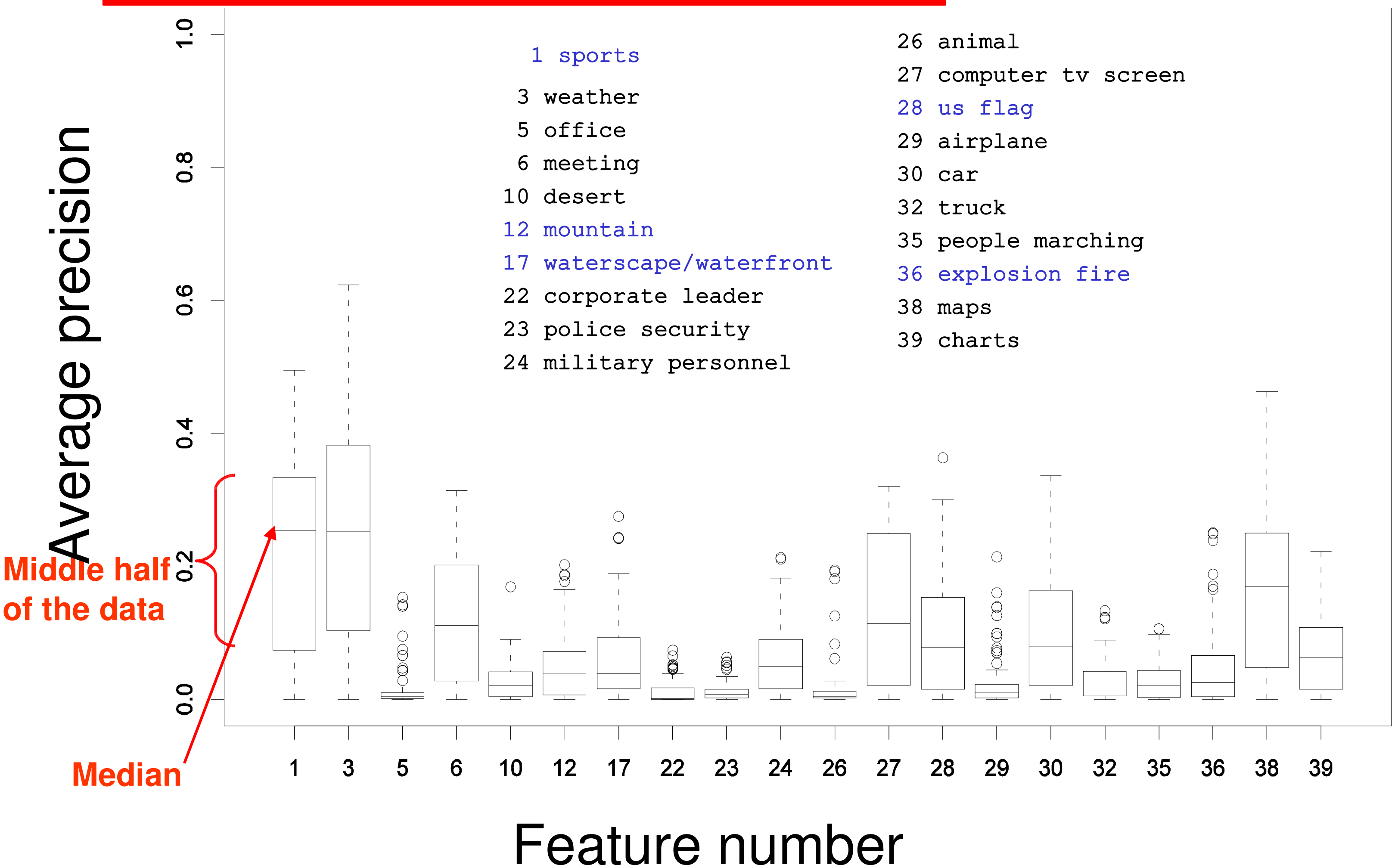
Category B results



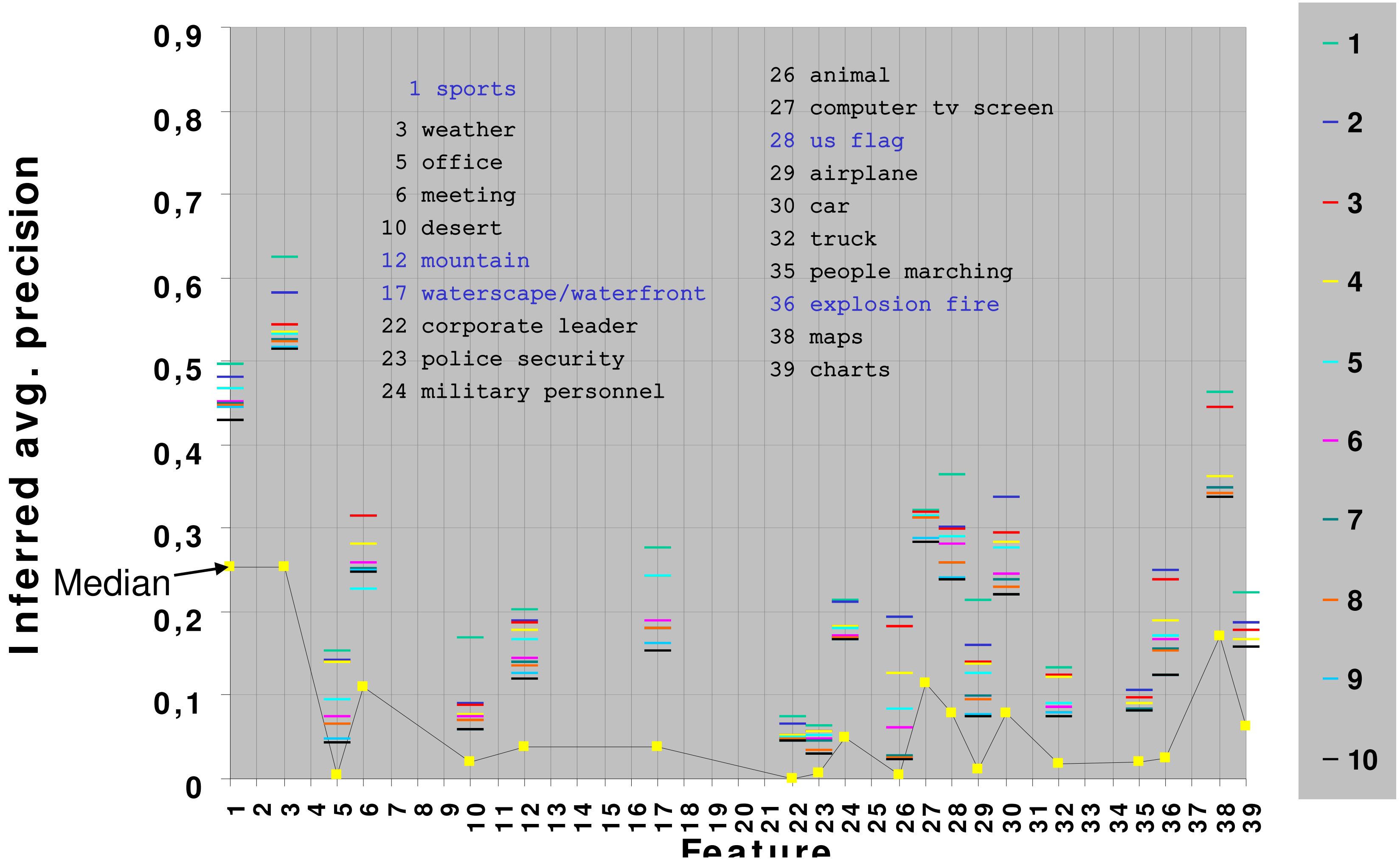
Category C results



Inferred Avg. Precision by feature (all runs)



Inferred avg. precision by feature (top 10 runs)



Which, if any, differences are significant, i.e. not due to chance?

Randomization testing

- Method of testing for significant pairwise differences between runs
  - Developed c.1935 by R.A. Fisher as thought experiment
  - Gained new usefulness with advent of computer intensive methods in statistics
  - Avoids dependence on (usually untrue) assumptions that samples are truly random, normally distributed, have equal variances, etc.
  - But makes no claims about populations

Randomization test procedure

1. Given observed scores for two systems on the same 20 features, calculate the mean score for each system and the observed difference of between the means.
2. Would like to know if the difference is due to the systems or to chance.
3. Generate a distribution of differences between the means under the null hypothesis that the difference is due to chance: for any feature, score from one system could equally likely have come the other
   - Calculate within feature pairwise difference & difference in means, once
   - For ~10,000 iterations or more
     - For each pair of scores, randomly change the sign of the difference
     - Sum the differences, calculate new mean, add it to the H° distribution
4. Count how many differences in H° are equal to or more extreme than the observed difference
5. Take [count / total number of generated differences] as probability (p) that the observed difference in means is due to chance.

Randomization test procedure

- Given observed scores for two systems on the same 20 features, calculate the mean score for each system and the observed difference of between the means.

```
R1:              0.467 0.434  0.013  0.314  0.041  0.188  0.242 ...
R2:              0.367 0.515  0.004  0.236  0.057  0.087  0.054 ...
(R1-R2)/20: SUM(+0.1  -0.081 +0.009 +0.078 -0.016 +0.101 +0.188 ...)/20
    = 0.033
```

- Generate a distribution of differences between the means under the null hypothesis that the difference is due to chance: for any feature, score from one system could equally likely have come the other

```
1.      SUM(-0.1 -0.081 -0.009 +0.078 -0.016 +0.101 +0.188 ...)/20 = -0.008
2.      SUM(+0.1 -0.081 +0.009 -0.078 +0.016 +0.101 -0.188 ...)/20 =  0.019
3.      SUM(-0.1 -0.081 -0.009 +0.078 -0.016 -0.101 +0.188 ...)/20 =  0.046
...
5.    SUM(+0.1 +0.081 +0.009 -0.078 +0.016 +0.101 +0.188 ...)/20 = -0.224
```

- 3145 of 95344 generated differences >= 0.033
- Probability observed difference is due to chance (p) = 0.03299

Significant differences among top 10 A-category runs (using randomization test, $p < 0.05$)

Run name (mean infAP)

- * A_tsinghua_6 (0.192)
- = A_IBM.MBWN_5 (0.177)
- = A_IBM.MRF_2 (0.176)
- = A_IBM.MAAR_3 (0.170)
- = A_IBM.MBW_1 (0.169)
- > A_CMU.Return..._6 (0.159)
- > A_IBM.UB_4 (0.155)
- > A_CMU.The_Empire..._5 (0.153)
- > A_CMU.A_New_Hope..._4 (0.148)
- > A_CMU.Attack of the..._2 (0.146)

A_tsinghua_6
- ➘ A_IBM.UB_4
- ➘ A_CMU.Return_of_The_Jedi_6
  - ➘ A_CMU.A_New_Hope_4
- ➘ A_CMU.The_Empire_Strikes_back_5
  - ➘ A_CMU.Attack_of_The_Clones_2

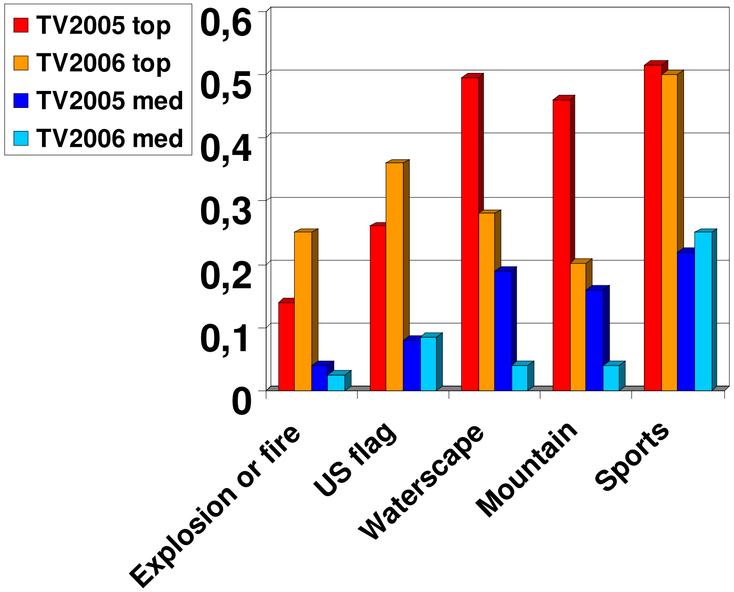
A_IBM.MRF_2
- ➘ A_CMU.Attack_of_The_Clones_2
- ➘ A_IBM.UB_4

A_IBM.MBWN_5
- ➘ A_CMU.Attack_of_The_Clones_2
- ➘ A_IBM.UB_4

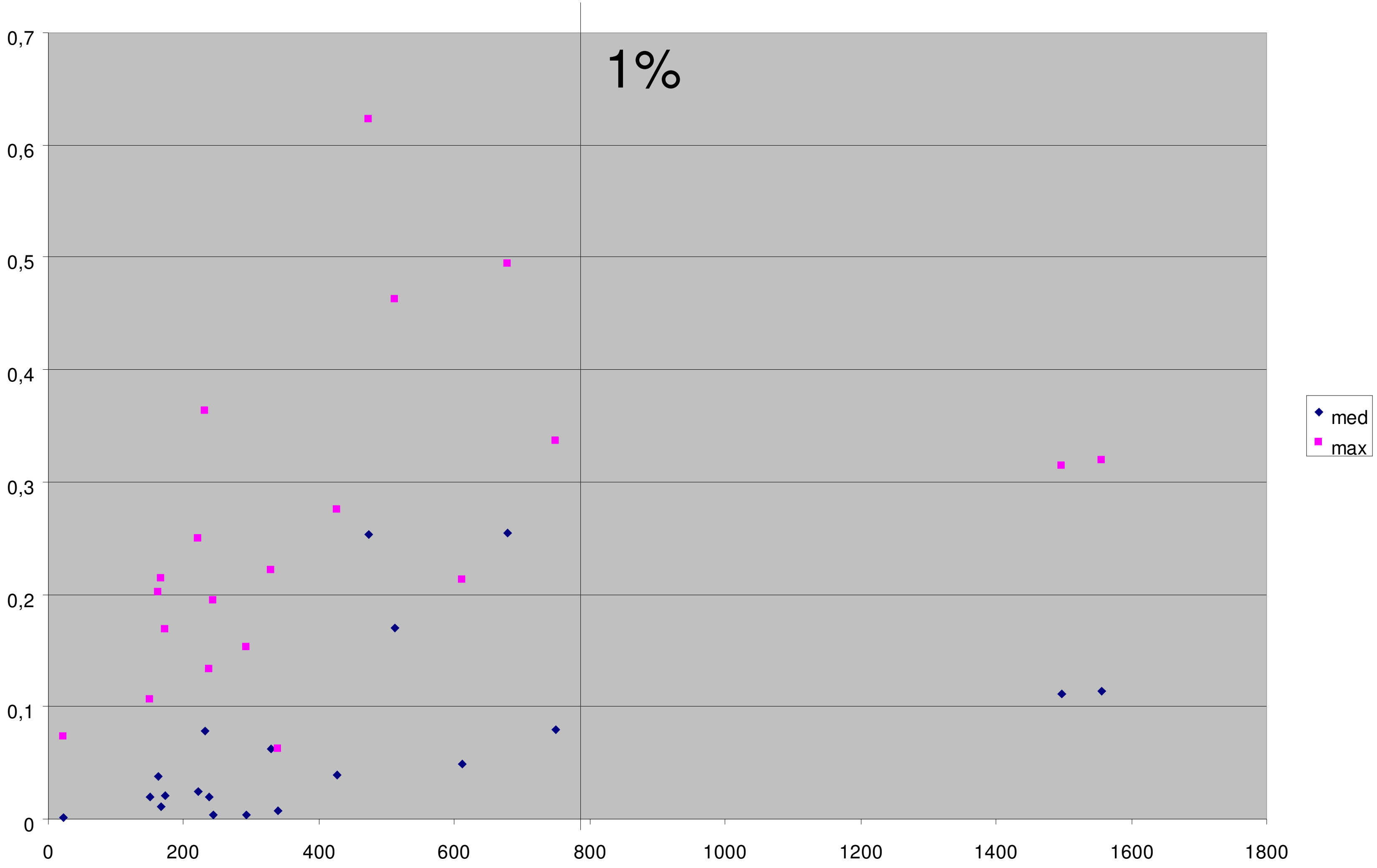
A_IBM.MAAR_3
- ➘ A_IBM.UB_4

Comparison with TV2005

- Some features were also evaluated last year
- Comparison yields mixed bag:
  - 2 features decreased
  - 2 features inceased
  - 1 feature stable
  - most of these features have just 100-200 true hits in the sampled pool
- Caveat: comparison is just indicative…
  - compare m.a.p and InfApp
  - but test set drawn from similar dataset as TV2005
  - Did anyone re-run last year's system?

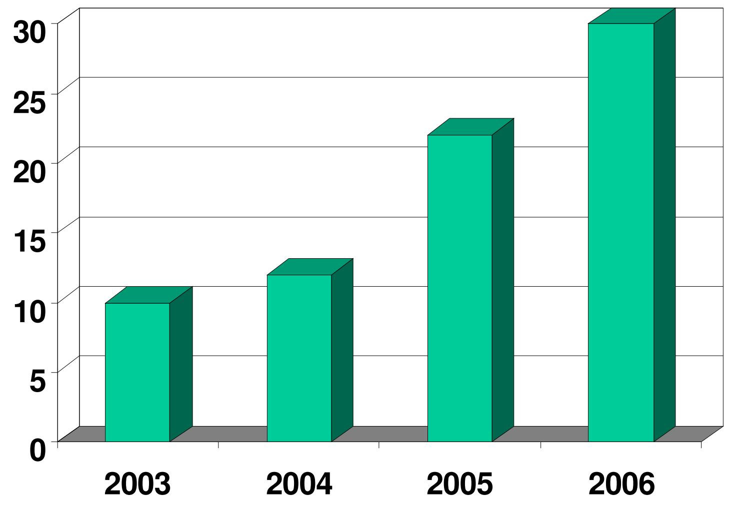


infAP vs. # true shots in test data



General observations (1)

- Participation is still increasing



- Maintained focus on cat A
- Most groups built a generic feature detector
- Top scores come from the usual suspects plus a few new groups

General observations (2)

- Many interesting new techniques are tried
- Some consolidation: SVM is the dominant classifier with robust results
- Good systems combine representations at multiple granularities
  - Salient point representation gaining more ground
- Good systems combine different feature types (c,t,e/s,a,T,f)
- 8/30 teams look at more than just the shot keyframe

- Many interesting multimodal/concept fusion experiments, room for more exploration here
- multi-concept fusion still of limited use (due to small lexicon?)
  - CMU: not many concepts support each other
  - Columbia: 3 out of 4 predicted concepts have 30% increase
- Can concept fusion learn from IR co-occurrence techniques?

Overview of approaches across sites

- feature types
  - c: color, t: texture, s:shape, e:edges, a:acoustic, f:face, T: text
- granularity (local, region, global)
- classifier techniques
- fusion
- generic vs. feature specific
- focus of site experimented marked in blue, speaking slots in yellow

Cat.	run tag best run	best	repr. granularity	features	temporal analysis	classifier	multimodal fusion	multiconcept fusion	eneric ?
A	tsinghua	0,192	global,grid, segm. point	c,t,T,f	camera motion, motion act.	svm	weight-select, rankboost, stackedSVM	tackedSVM, rules	
A	IBM.MAAR	0,170	?	?		svm,?	?	?	
A	CMU.A_New_Hope	0,148	rid (5x5) +points	,t,T		svm	logistic regression, early, late, borda	multi discr RF (chi2 selection)	
A	COL1	0,142	SIFT points/grid	c,t,T	MD	vm	average fusion	oosting CRF (PMI selection)	
A	ucb_1best	0,122	points	,e,T	shot context	svm	svm	svm	
A	UCF.CE.PROB	0,119		c,e		svm	average/product/KDE		
B	MM.bottom	0,117	global, grid, point			svm/ log reg / LD	early/ late fusion	svm	
A	KSpace-base	0,110	grid	c,t,e,T	camera motion	svm	bayesian (DS)		gen+specific
A	CityUHK1	0,106	points+grid	c,t	EMD	svm	average fusion		
A	MSRA_TRECVID	0,086	global, grid	c,t,s,f,T		SVM, KDE, manifold ranking, t-graph	weighted fusion, also looked at unlabeled data		
A	NTU	0,073							
B	PicSOM_F7	0,064	grid	c,t,T	motion act. average c,t, for shot	SOM	linear combination	handpicked negative concepts	
B	FXPAL-06Beta	0,059	MM	MM		svm		DRF / chi2	
B	XVGG_A	,053	points (sparse/dense)	c,e,f		SVM	Borda Count		

C at.	run tag best run	be st	repr. granularity	feature s	temporal analysis	classifier	multimodal fusion	multiconce pt fusion	gen eric?
A	i2Rnus	0,040	grid	c,t,T	frame clustering, bigrams	SVM,LDF,GMM		cond prob	
A	NII_ISM_R1	0,033	overlapping grid	loc. bin. pat.		SVM			
B	clips.local-reuters-kernel-prod	0,031	local+global	c,t,T		SVM			
A	TokyoTech1	0,0 30							
A	ZJU	0,029	global	c,t,e,T,a		VM	ultimodal subspace correlation propag		
C	kddi.SiriusCy3	0,0 26	grid + points	s		Haar/KNN			not all
A	ilkent1	,021	rid	,t,e,T		NN			
B	TZI_Avg	0,0 21		c,T,e,f, a	every 20th frame	SVM	weighted average, prob. relax. labelling	cond prob	+sp ecific
A	UEC_Common	0,0 06							
A	Glasgow.Sheffield 01	0,0 05		T		tfidf			
A	LIP6.FuzzyDT	0,0 04	grid	p,c		fuzzy decision trees			
A	UR01-SVM	0,0 02	points	c,t		SVM		NN	
A	FD_SCM_BN	0,0 01	points	c,t		GMM/SVM		cond. P	
A	icl.jhu_4	,001	rid	,t,T		likelihood ratio (HMM)	source adaptation		
C	Iowa06FE01	,001							
A	COST292R1	0,000	points/grid/LSA	c,t / T		NN/Bayes			ot all

Issues

- How to make the most of a fixed limited number of assessor time
  - Sampling method
  - Equal pool size for each feature?
- Repetition of advertisement clips was less of an issue as in TV2005
- Systematic study of interaction between search and HLF
- How to proceed after 5 years of HLF?
  - massive scaling requires massive amounts of annotation and assessment time

Discussion input

- How to make the most of a fixed limited number of assessor time
  - Sampling method refinement
    - top->sample->unique vs. top->unique-sample?
    - mark ignore vs. mark non relevant
  - map vs. precision@N
  - Equal pool size for each feature?
- How to proceed after 5 years of HLF?
  - massive scaling requires massive amounts of annotation and possibly assessment time
  - Explore social tagging, annotation as a game?