INAOE's participation at PAN'13: Author Profiling task

A. Pastor López-Monroy, M.Sc.[1]

M. Montes-y-Gómez, Ph.D.[1] **H. J. Escalante, Ph.D.**[1]

L. Villaseñor-Pineda, Ph.D.[1] **E. Villatoro-Tello, Ph.D.**[2]

September-2013

México

Computer Science Department,
Instituto Nacional de Astrofísica, Óptica y Electrónica [1]

Information Technologies Department,
Universidad Autónoma Metropolitana-Cuajimalpa [2]

Contents

Introduction

- The Author Profiling (AP) task consists in knowing as much as possible about an unknown author, just by analyzing a given text [5].
- Initially some works in AP have started to explore the problem of detecting gender, age, native language, and personality in several domains [5, 9, 1].
- One of the domains of interest is the social media data (e.g., blogs, forums, reviews, tweets, chats, etc.).
- The PAN13 AP task consists in profiling **age** and **gender** in social media data.
- The AP task can be approached as a classification problem, where profiles represent the classes to discriminate.

The challenging raw social media data

There are some known issues that could pose a problem to the effectiveness of most common/standard techniques in text mining:

- **Sparsity**:
  - Short texts (e.g., comments, reviews): there are few terms in each of them to take that as a valuable evidence.
  - Large sets of documents: where normally exist huge vocabularies (standard and non-standard).

- **Noise in the data**:
  - The easiness to write and sent messages leads to make spelling/grammatical mistakes.
  - Slang vocabulary.
  - Noise in the labels of documents.

Typical representation of documents

One of the most common approaches is the Bag of Terms (BOT)

Some shortcomings of BOT like representations are:

- They produce representations with high dimensionality and sparsity.
- They do not preserve any kind of relationship among terms.

Our proposal

We propose the use of very simple but highly effective meta-attributes for:

- Having different textual features (e.g., content, style) in term vectors that represents relationships with each profile.
- Representing documents using the latter term vectors to highlight the relationships with each profile.
- Facing problems like: high dimensionality, sparsity of vectors and the noisy in text data.

These attributes are inspired in some ideas from CSA [7] to represent documents in text classification.

Document Profile Representation

- DPR stores textual features of documents in a vector, where the problem of dimensionality is limited by the number of profiles to classify.
- DPR is built in two steps:
  - Building term vectors in a space of profiles.
  - Building document vectors in a space of profiles.
- Example of the final document-profile matrix:

	p_1	.	.	.	p_i
d_1	$dp_{11}(p_1, d_1)$	.	.	.	$dp_{i1}(p_i, d_1)$
.	.	.			
.	.		.		
.	.			.	
d_j	$dp_{1j}(p_1, d_j)$				$dp_{ij}(p_i, d_j)$

Term representation

For each term t_j in the vocabulary, we build a term vector $\mathbf{t_j} = \langle tp_{1j}, \ldots, tp_{ij} \rangle$, where tp_{ij} is a value representing the relationship of the term t_j with the profile p_i. For computing tp_{ij} first:

$$wtp_{ij} = \sum_{k:d_k \in P_i} \log_2 \left(1 + \frac{tf_{kj}}{len(d_k)}\right)$$

	p_1	.	.	.	p_i
t_1	$wtp_{11}(p_1, t_1)$	.	.	.	$wtp_{i1}(p_i, t_1)$
.	.	.			
.	.		.		
.	.			.	
t_j	$wtp_{1j}(p_1, t_j)$				$wtp_{ij}(p_i, t_j)$

Term normalization

So we get $\mathbf{t_j} = \langle wtp_{1j}, \ldots, wtp_{ij} \rangle$, and finally we normalize each wtp_{ij} as:

$$tp_{ij} = \frac{wtp_{ij}}{\sum_{j=1}^{TERMS} wtp_{ij}}$$

$$tp_{ij} = \frac{wtp_{ij}}{\sum_{i=1}^{PROFILES} wtp_{ij}}$$

In this way, for each term in the vocabulary, we get a term vector $\mathbf{t_j} = \langle tp_{1j}, \ldots, tp_{ij} \rangle$.

Documents representation

Add term vectors of each document. Documents will be represented as $\mathbf{d_k} = \langle dp_{1k}, \ldots, dp_{nk} \rangle$, where dp_{ik} represents the relationship of d_k with p_i.

$$\vec{d}_k = \sum_{t_j \epsilon D_k} \frac{tf_{kj}}{len(d_k)} \times \vec{t}_j$$

where D_k is the set of terms of document d_k.

	p_1	.	.	.	p_i
d_1	$dp_{11}(p_1, d_1)$	.	.	.	$dp_{i1}(p_i, d_1)$
.	.	.			
.	.		.		
.	.			.	
d_j	$dp_{1j}(p_1, d_j)$				$dp_{ij}(p_i, d_j)$

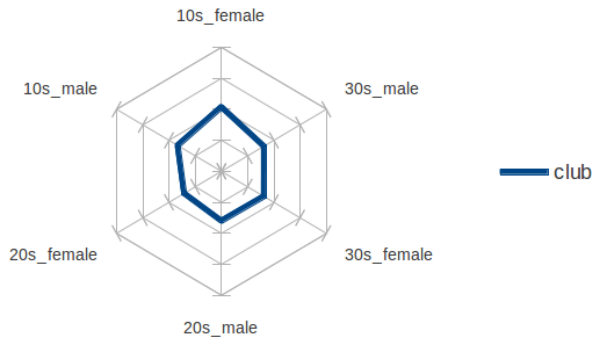
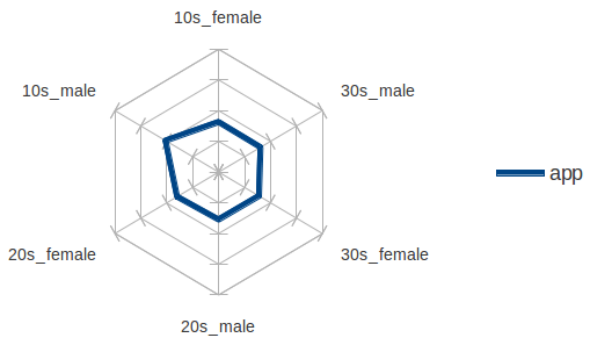
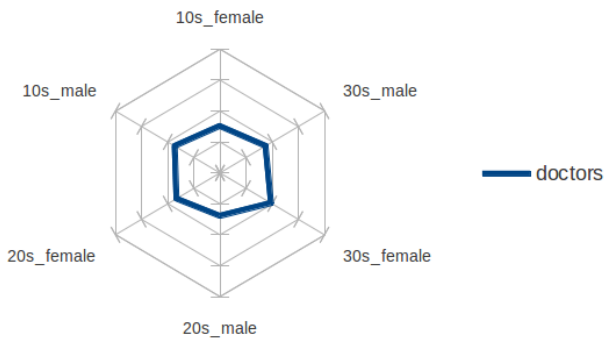
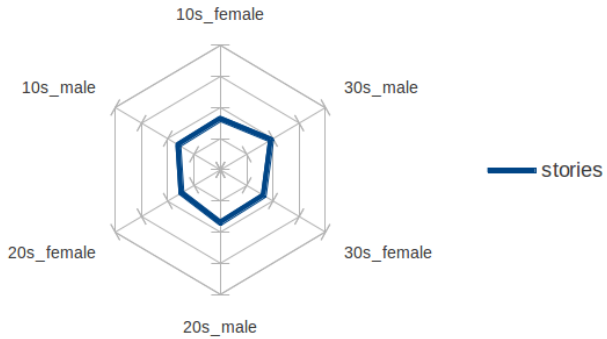
Summary of Document Profile Representation

The representation is built in two steps:

- Building term vectors that represents relationships among profiles.
- Building document vectors that represents relationships among profiles.

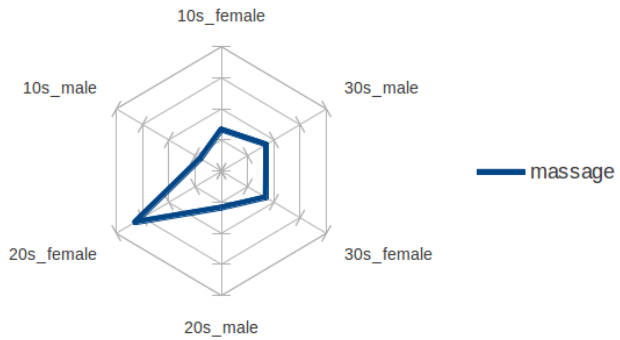
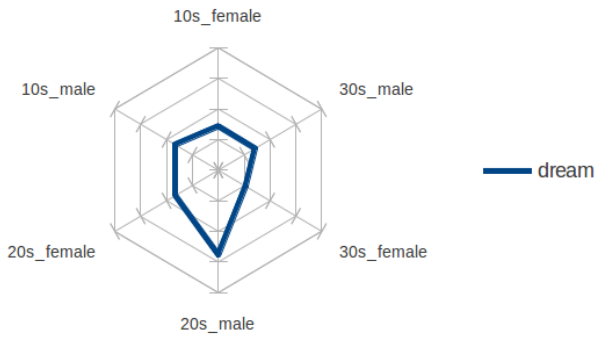
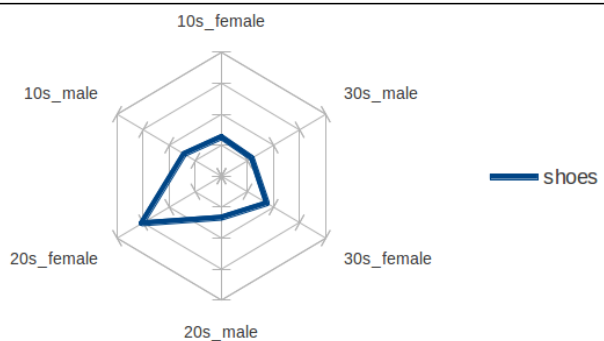
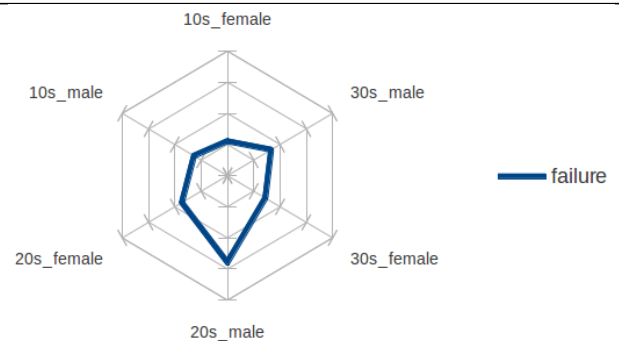
In the following slides we show some examples of how looks some high descriptive term vectors.

Examples of high descriptive term vectors.

Good for profile **"10s-female"**	Good for profile **"10s-male"**
	
similar: birds, amazing, mom, plant, injuries	**similar:** aids, classes, hardware, trend
Good for profile **"30s-female"**	Good for profile **"30s-male"**
	
similar: pleasant, long-term, heat, accurate	**similar:** dollar, satisfaction, power, drug

Examples of high descriptive term vectors.

Some term vectors have stronger peaks.

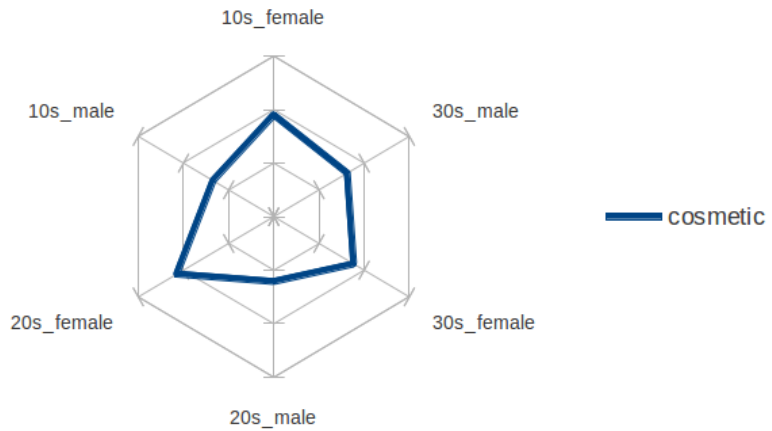
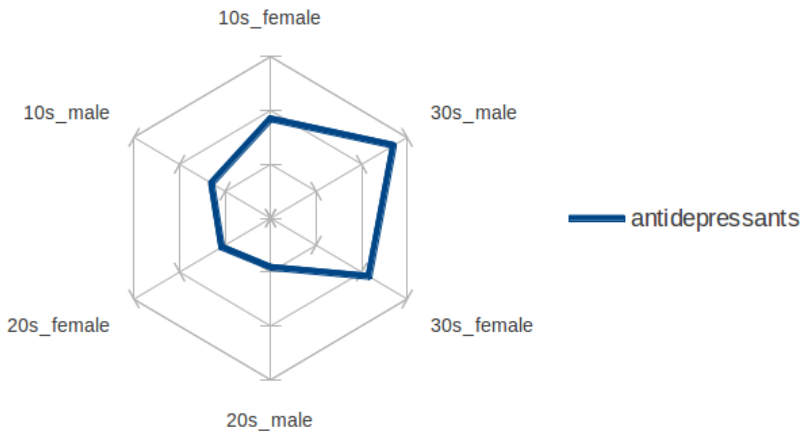
Good for profile **"20s-female"**	Good for profile **"20s-male"**
	
	
similar: flowers, dresses, nike, mulberry, noise	**similar:** wise, golden, trust, loose, nice

Term vectors for multiple relationships observations

There are some term vectors that show a strong peak for two or three profiles. They are also highly descriptive term vectors for predicting for example:

- age
- gender
- specific age females
- specific age males

Examples of term vectors for multiple relationships observations

Good for profile **"female"**	Good for profile **"30s"**
	

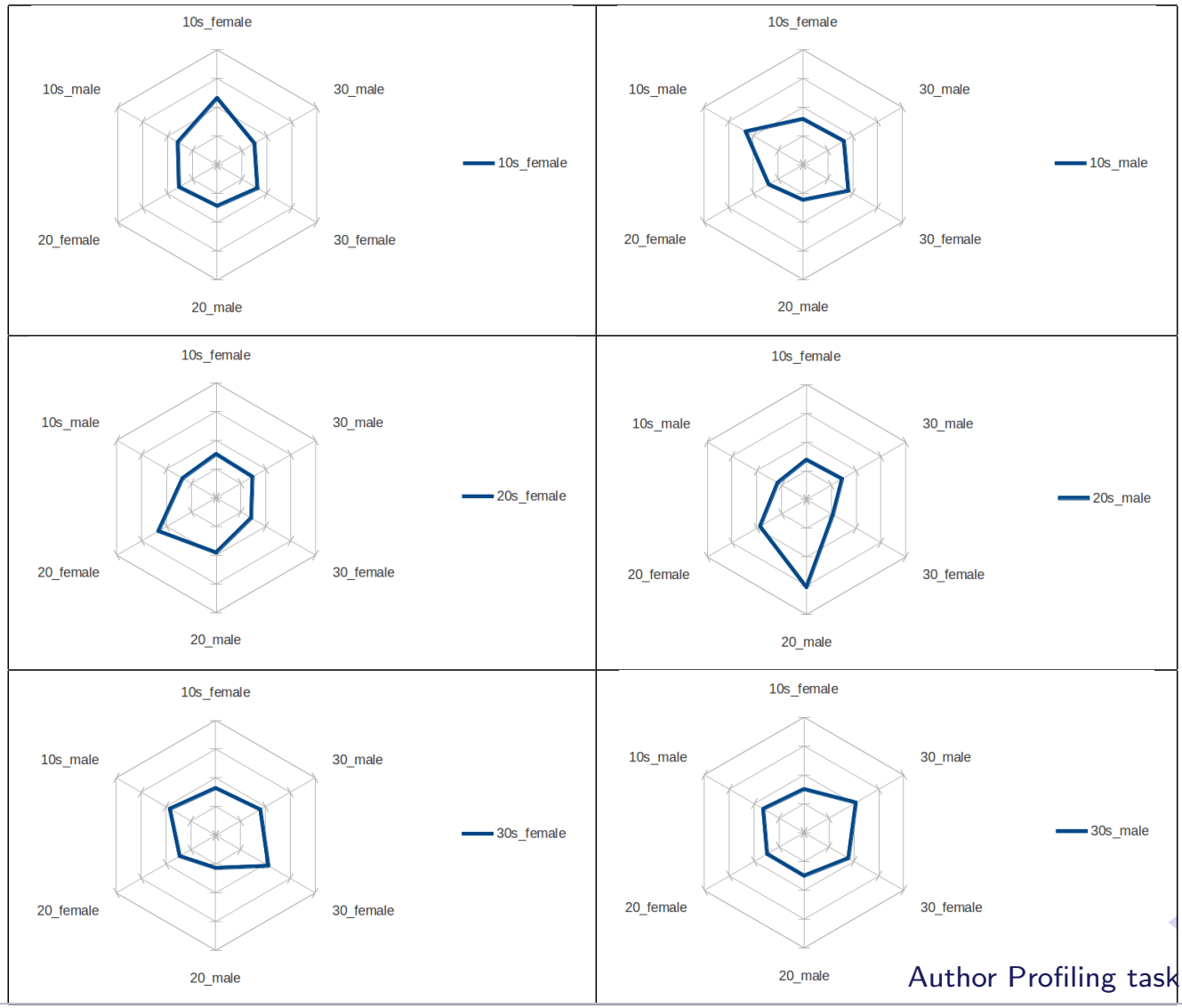
There are other similar term vectors for specific profiles for example:

- ":)": for detecting young people (e.g. profiles 10s, and 20s).
- "game": for the prediction of males.

Vectors for profile relationships

- Some of the latter terms had already been identified in the literature [5, 9, 1] for AP.
- Having such terms represented with high level attributes lets us know the meaningful relationships they keep with other profiles.
- A document vector is built through the summation of its term vectors.
- In the next slide we show the document centroids for each profile.

Document centroids for each profile



Evaluation

- We approached the AP task as a six *age-gender* profiling classes: *10s-female, 10s-male, 20s-female, 20s-male, 30s-female, 30s-male*.
- Although some other works have approached separately the Age and Gender detection, the relationships between age-gender profiles could be important [8].
- From the point of view of text classification, we have a set of training documents for each category (e.g., 10s-female and 10s-male. etc.).

Evaluation

Description of the corpus according to our used textual features **(words, stopwords, punctuation marks and emoticons).**

Description for the English corpus							
		Statistics by category					
criteria	**Total**	10s-f	10s-m	20s-f	20s-m	30s-f	30s-m
authors	236600	8600	8600	42900	42900	66800	66800
mean	1058.11	1118.91	1169.02	1005.92	822.75	1172.32	1106.46
std	872.69	918.03	717.56	786.67	918.92	696.84	1021.10
min	1	1	1	1	1	1	1
25 %	591	669	692	367	75	701	637
50 %	898	987.5	1176	845	685	1213	959
75 %	1541	1553	1577.25	1535	1434	1567	1557
max	69374	33566	12791	19308	51453	50077	69374

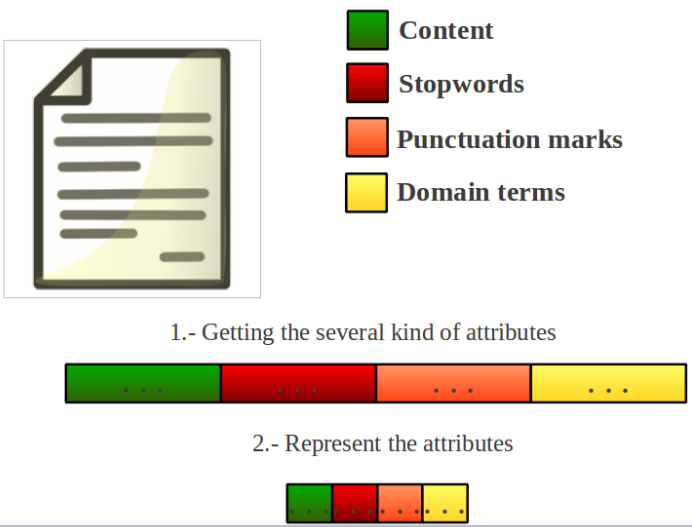
Evaluation

Description of the corpus according to our used textual features **(words, stopwords, punctuation marks and emoticons).**

Description for the Spanish corpus							
		Statistics by category					
criteria	**Total**	10s-f	10s-m	20s-f	20s-m	30s-f	30s-m
authors	75900	1250	1250	21300	21300	15400	15400
mean	374.19	234.60	255.36	369	349.044	376.71	434.58
std	704.23	586.42	664.79	586.82	719.41	630.95	884.97
min	1	3	1	1	1	1	1
25 %	32	33	21	42	31	30	25
50 %	87	74	53	116	79	80	71
75 %	376	212	174	410	323	403	447.25
max	26163	11629	12257	14507	26163	13869	16529

Evaluation

- To build the representation, a vocabulary of the 50,000 most frequent terms were considered. The considered terms belongs to four different modalities: i) content features, ii) stopwords, iii) punctuation marks, and iv) domain specific vocabulary (e.g., emoticons and hastags).
- The LIBLINEAR library was used to perform the prediction [4]. During the development period, we performed a stratified 10 cross fold validation using the training PAN13 corpus.



Final results

- Experiments using the Second-Order-Attributes (SOA) and Bag-of-Terms (BOT) computed over the 50,000 most frequent terms on the datasets.

Detailed classification accuracy											
	Training data				*Test data*			*Averaged results for all participants*			
	SOA			BOT	SOA			AVG			
	Gender	Age	Total	Total	Gender	Age	Total	Gender (st.dv.)	Age (st.dv.)	Total (st.dv.)	
English	61.3	63.7	**41.9**	36.6	56.90	65.72	**38.13**	53.76 (3.33)	53.51 (12.50)	28.99 (7.42)	
Spanish	70.5	72.7	**54.8**	41.9	62.99	65.58	**41.58**	55.41 (4.99)	49.04 (14.15)	27.67 (9.35)	

Top 10 ranking in the PAN13

Submission	Accuracy			Runtime
	Total	Gender	Age	(incl. Spanish)
meina13	0.3894	0.5921	0.6491	383821541
pastor13	0.3813	0.5690	**0.6572**	**2298561**
mechti13	0.3677	0.5816	0.5897	1018000000
santosh13	0.3508	0.5652	0.6408	17511633
yong13	0.3488	0.5671	0.6098	577144695
ladra13	0.3420	0.5608	0.6118	1729618
ayala13	0.3292	0.5522	0.5923	23612726
gillam13	0.3268	0.5410	0.6031	615347
kern13	0.3115	0.5267	0.5690	18285830
haro13	0.3114	0.5456	0.5966	9559554
baseline	0.1650	0.5000	0.3333	–

Submission	Accuracy			Runtime
	Total	Gender	Age	(incl. English)
santosh13	0.4208	0.6473	0.6430	17511633
pastor13	0.4158	0.6299	**0.6558**	**2298561**
haro13	0.3897	0.6165	0.6219	9559554
flekova13	0.3683	0.6103	0.5966	18476373
ladra13	0.3523	0.6138	0.5727	1729618
jimenez13	0.3145	0.5627	0.5429	3940310
kern13	0.3134	0.5706	0.5375	18285830
yong13	0.3120	0.5468	0.5705	577144695
ramirez13	0.2934	0.5116	0.5651	64350734
aditya13	0.2824	0.5000	0.5643	3734665
baseline	0.1650	0.5000	0.3333	–

Conclusions

1. The proposed approach is the best method at PAN'13 to predict age profiles in blogs (for both corpora).
2. For the six-class AP task at PAN'13, our results overcomes the conventional BOT and holds the first position for both languages (overall accuracy), and second position for each one.
3. For the english corpus, the proposed approach took only 0.22 % (more than 454 times faster) of the time required by the method in one position below, and 0.59 % (more than 166 times faster) of the time required by the method in first position.
4. This is the first time that AP is addressed using attributes that represent relationships with profiles.
5. Through very low computational cost our proposal can build discriminative low dimensional dense vectors for AP

Thank you

. . . Questions?

References

Shlomo Argamon, Moshe Koppel, James W Pennebaker, and Jonathan Schler.
Automatically profiling the author of an anonymous text.
Communications of the ACM, 52(2):119–123, 2009.

Federica Barbieri.
Patterns of age-based linguistic variation in american english1.
Journal of Sociolinguistics, 12(1):58–88, 2008.

Penelope Eckert.
Age as a sociolinguistic variable.
The handbook of sociolinguistics, 151:67, 1997.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin.
LIBLINEAR: A library for large linear classification.
Journal of Machine Learning Research, 9:1871–1874, 2008.

Moshe Koppel, Shlomo Argamon, and Anat Rachel Shimoni.
Automatically categorizing written texts by author gender.
Literary and Linguistic Computing, 17(4):401–412, 2002.

William Labov.
The intersection of sex and social class in the course of linguistic change.
Language variation and change, 2(2):205–254, 1990.

Zhixing Li, Zhongyang Xiong, Yufang Zhang, Chunyong Liu, and Kuan Li.
Fast text categorization using concise semantic analysis.