Modern Authorship Attribution

Efstathios Stamatatos

University of the Aegean

Outline

- Authorship analysis tasks
- Author identification characteristics
- Dealing with limited training data
  - Tensor models (ECAI'08 poster)
- Conclusions

Authorship Analysis Tasks

- Author identification
  - Given a set of candidate authors, to attribute a text to one of them
- Author verification
  - Given texts of a certain author, to decide whether an unseen text was written by that author or not
- Author profiling or characterization
  - Extracting information about the age, gender, dialect etc. of the author
- Plagiarism detection
  - Determining whether a given document was produced by copying or including another author's ideas or writing without proper acknowledgement of the original source

Author Identification

- The assignment of a text of unknown authorship to one author, given
  - a set of candidate authors
  - text samples of undisputed authorship for each candidate author
- A multi-class single-label text categorization task
- It can be applied to e-mail messages, online forum messages, blogs, source code, etc.
- Applications in areas such as intelligence, criminal law, computer forensics, etc.

Characteristics of Author Identification

- Frequency is the most important factor for selecting features
  - in topic-based TC the most frequent words are excluded
- Shortage of training texts for the candidate authors
  - amount of training texts
  - length of training texts
- Imbalanced training data

A State-of-the-art Approach

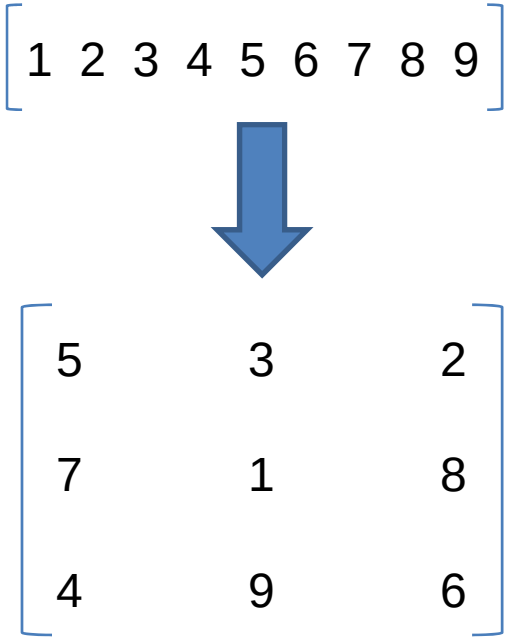
- Stylometric features: Character *n*-grams
  - Character 3-grams found to work well for English
  - High dimensionality (thousands of features)
- Classifier: A linear SVM model

How to deal with Limited Training Data

- Try to enrich the training set
- Use more effective classification algorithms
- Use more effective representations

- Proposed by (Plakias & Stamatatos, 2008):
  - Use a tensor space representation instead of a traditional vector space representation
- Why?
  - A tensor model requires much less parameters to be learnt

From Vectors to Tensors



Vector vs. Tensor Space Representation

- Vector space:
  - a text is a vector in R^n, where n is the number of features
  - a linear classifier (e.g., SVM) is $\mathbf{a}^T\mathbf{x}+b$, that is, there are $n+1$ parameters to be learnt (b, a_i, $i=1,\ldots,n$)
- Tensor space (second order)
  - a text is a matrix in $R^x \otimes R^y$, where x and y are the dimensions of the matrix
  - a vector $\mathbf{x} \in R^n$ can be transformed to a second order tensor $\mathbf{X} \in R^x \otimes R^y$ provided $n \approx x*y$
  - a linear classifier in $R^x \otimes R^y$ is $\mathbf{u}^T\mathbf{X}\mathbf{v}+b$, that is, there are $x+y+1$ parameters to be learnt (b, u_i, $i=1,\ldots y$, v_j, $j=1,\ldots x$)
  - the number of parameters is minimized when $x=y$ ($<< n$)

How to Use Tensors

- We need a classification algorithm able to handle tensors
  - Support Tensor Machines (Cai, et al., 2006) is an extension of SVM
  - Iteratively computes **u** and **v**
  - Much slower than SVM
- We need a method to fill the matrix
  - The position of each feature in the matrix is now important
  - Each feature is strongly associated with features of the same row and column

Feature Relevance

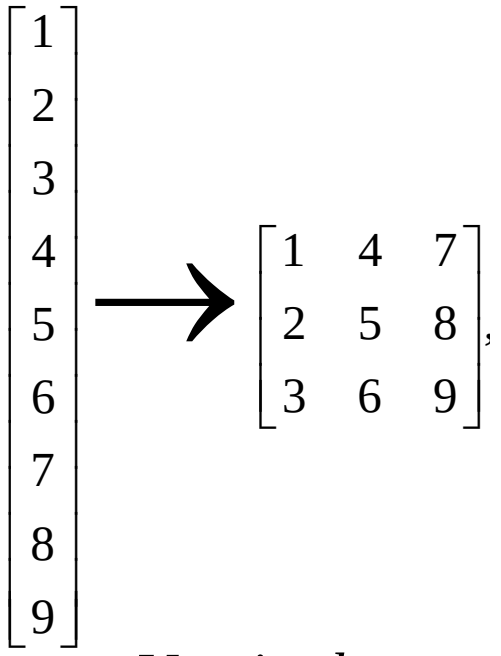
- In a binary classification case, where we want to discriminate author A from author B, the relevance $r(x_i)$ of a feature x_i is

$$r(x_i) = \frac{f_A(x_i) - f_B(x_i)}{f_A(x_i) + f_B(x_i) + b}$$

- The higher the $r(x_i)$, the more important the feature x_i for author A
- b is a smoothing factor
- The feature vector is sorted by feature relevance

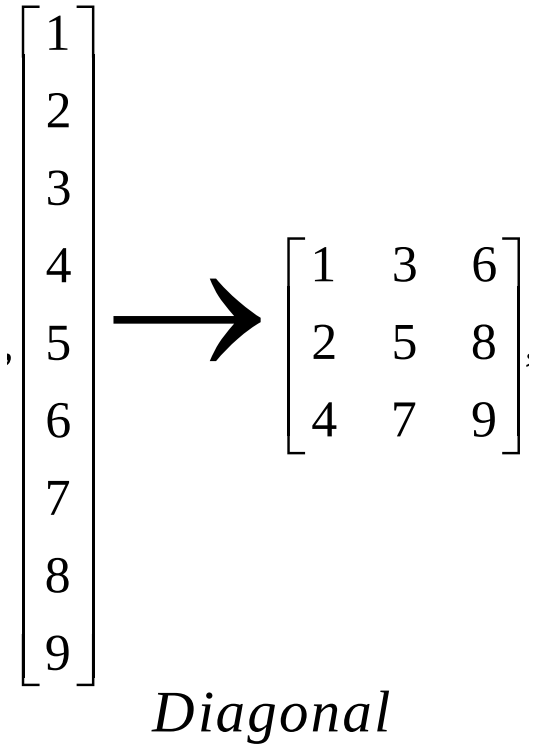
Matrix Filling: Vertical

- The columns of the matrix are filled with decreasing relevance values
- The first columns of the tensor will be strongly associated with author A and the last columns with author B
- The rows of the matrix contain features of mixed importance for the two authors



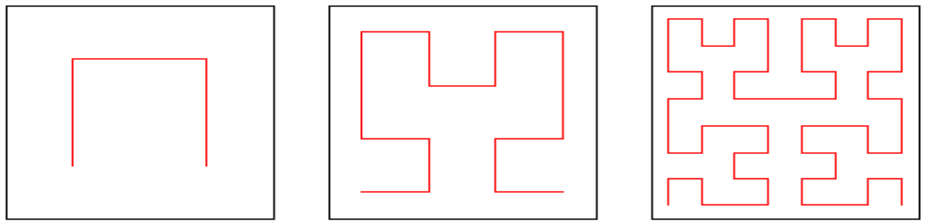
Matrix Filling: Diagonal

- We start from the upper left corner of the matrix and fill diagonals with decreasing relevance values
- The first rows and columns are mainly associated with author A while the last rows and columns with author B



Matrix Filling: Hilbert

- We use the Hilbert space filling curve
- This technique produces small neighbourhoods of relevant features
- Any row or column contain features of mixed importance



$$\begin{bmatrix}1\\2\\3\\4\\5\\6\\7\\8\\9\end{bmatrix} \rightarrow \begin{bmatrix}4 & 3 & 2\\5 & 6 & 1\\8 & 7 & 9\end{bmatrix}$$

Hilbert

Experiments: Corpus

- Newswire stories in English (RCV1)
- 10 authors
- All texts are under topic class CCAT (about corporate and industrial news)
- Three versions of this corpus were formed using 50, 10 or 5 training texts per author
  - In all cases, the test corpus comprises 50 texts per author not overlapping with the training texts

Experiments: Settings

- Text representation:
  - the 2,500 most frequent 3-grams of the training corpus
- Tested models
  - A linear SVM model using the vector of 2,500 features (C=1)
  - A STM model based on a 50x50 matrix (C=0.1) using
    - no matrix filling strategy
  - A STM model based on a 50x50 matrix (C=0.1, b=1) using
    - vertical,
    - diagonal, or
    - Hilbert space filling

Experiments: Results

Method	Training texts per author		
	50	**10**	**5**
SVM	**80.8%**	64.4%	48.2%
STM-Simple	70.4%	54.4%	44.2%
STM-Vertical	78.0%	**68.0%**	**51.2%**
STM-Diagonal	75.6%	60.8%	47.6%
STM-Hilbert	76.6%	66.6%	46.0%

Conclusions

- Matrix filling methods improve the results
- Results are promising for the proposed models when dealing with limited training data
  - The training time cost is significantly higher
- SVM is superior when multiple training texts are available
- The vertical matrix filling method seems to perform better
  - This method produces some subsets of features (columns) that are strongly associated with the authors as well as other subsets (rows) that contain features of mixed importance for the authors
  - Further experiments need to verify this

Challenges

- Training texts
  - Dealing with limited and imbalanced training data
- Text-length
  - Is there a threshold to capture style adequately?
- Style vs. Topic
  - Low level features also capture thematic information
- Is it possible to explain the stylistic differences?
  - Very important in court process
  - Association of low-level and high-level features
- Inter-genre models
  - Is a model robust enough to be trained with texts on one genre and identify the author of texts on a different genre?