Novel representations and methods in text classification

Manuel Montes, Hugo Jair Escalante

Instituto Nacional de Astrofísica, Óptica y Electrónica, México.

http://ccc.inaoep.mx/~mmontesg/

http://ccc.inaoep.mx/~hugojair/

{mmontesg, hugojair}@innoep.mx

7th Russian Summer School in Information Retrieval
Kazan, Russia, September 2013
ENHANCING THE BOW WITH SEQUENTIAL INFORMATION
Outline

• Bag of words

• Extensions to incorporate sequential information
  – Maximal frequent sequences
  – Sequential patterns
  – The LOWBOW framework

• Text categorization under LOWBOW

• Authorship attribution with LOWBOW
Bag of words

• Under the bag-of-words framework a document is represented by the **set of terms** that appear in it.

• By definition, BOW is an orderless representation.
Bag of words

- Under the bag-of-words framework a document is represented by the **set of terms** that appear in it.

- By definition, BOW is an orderless representation.

**Yo me río en el baño**
(I am laughing at the bathroom)

**Yo me baño en el río**
(I am taking a shower at the river)

Same BoW representation
**different meaning**
Bag of words

• There have been several efforts trying to incorporate sequential information into BoW-based representations
  – **Ngrams**: Terms are defined as sequences of characters or words
  
  – **Maximal frequent sequences**: Frequent sequences of words are discovered (with/without gaps)
  
  – **Phrase patterns**: Sequential data mining is applied to detect sequential patterns (with gaps)
  
  – **Methods based on linguistic analyses**: POS tagging, syntactic trees, etc.
Bag of Ngrams

• An Ngram is a sequence of N-terms (e.g., words / characters):
  – Russian-federation / bag-of-words / in-god-we-trust ...
  – the / mex / lol / wtf ...

• A sliding window is applied to the documents, all Ngrams found in the corpus form the vocabulary

• Documents are represented by the bag of Ngrams that they contain

**Document:** Russian Summer School in Information Retrieval

<table>
<thead>
<tr>
<th>Unigrams</th>
<th>Bigrams</th>
<th>Tri-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>Russian, Summer School, in, Information, Retrieval</td>
<td>Russian-summer, summer-school, school-in, in-information, information-retrieval</td>
<td>Russian-summer-school, Summer-School-in, School-in-Information, in-Information-Retrieval</td>
</tr>
</tbody>
</table>
Bag of Ngrams

- An Ngram is a sequence of N-terms (e.g., words / characters):

- Ngrams capture low-range sequential information
- Fixed length patterns (usually n≤5);
- Satisfactory results have been reported in non-thematic tasks
- The size of the vocabulary increases dramatically
- When using characters, they can capture style aspects
- No significant improvements over standard BOW
Bag of Ngrams

• An Ngram is a sequence of N-terms (e.g., words / characters):

  ✓ Ngrams capture low-range sequential information
  ✗ Fixed length patterns (usually n≤5);

**Skyp-grams**: Extension to Ngrams that allows us to consider gaps between terms to build Ngrams. Example:

Russian Summer School in Information Retrieval

<table>
<thead>
<tr>
<th>Bigrams</th>
<th>2-skyp-bigrams</th>
</tr>
</thead>
<tbody>
<tr>
<td>Russian-summer, summer-school</td>
<td>Russian-school, Russian-in, Summer-</td>
</tr>
<tr>
<td>school-in, in-information,</td>
<td>in, Summer-in, Summer-Information,</td>
</tr>
<tr>
<td>information-retrieval</td>
<td>School-information, in-retrieval</td>
</tr>
</tbody>
</table>

Increases the range of sequential information, but augments the vocabulary size
**Maximal frequent sequences**

- Each document is seen a sequence of words (items)

- The goal is to identify *interesting* sequences of words that can be used to characterize documents, e.g.:

  Russian-School-Information-Retrieval

- No fixed-length constraints are imposed (as in n-grams)

- Reduce overlapping information in the representation

- Gaps are allowed in sequences
Maximal frequent sequences

• Definitions:
  – A sequence $p = p_1, \ldots, p_k$ is a subsequence of another sequence $q = q_1, \ldots, q_m$ if all of the items $p_i, 1 \leq i \leq k$, occur in $q$ and they occur in the same order as in $p$
  – A sequence $p$ is frequent in document collection $D$ if $p$ is a subsequence of at least $\sigma$ documents in $D$
  – A sequence $p$ is a maximal frequent sequence in $D$ if there does not exist any sequence $p'$ in $D$ such that $p$ is a subsequence of $p'$ and $p'$ is frequent in $D$

• There are efficient algorithms to identify all of the MFS
MFS for authorship attribution

• Authorship attribution: Given texts of uncertain authorship and texts written by a set of candidate authors, the task is to map the uncertain texts onto their true authors among the candidates.

• Applications include: fraud detection, spam filtering, computer forensics and plagiarism detection
MFS for authorship attribution

- Instance-based approach to authorship attribution (~text categorization)

![Diagram showing the process of authorship attribution with training texts and classifier training leading to attribution model and most likely author determination.]
Maximal frequent sequences

• MSF for authorship attribution

Let $D_T$ be the set of labeled documents that will be used for training
Let $d$ be an anonymous document

**TRAINING**
1. Set the value of the frequency threshold $\sigma = 2$
2. Set the feature set $F_1 = \{\emptyset\}$
3. DO
   a. Enumerate all maximal frequent word sequences in $D_T$ corresponding to the frequency threshold $\sigma$. Name the set of sequences $S_\sigma$
   b. Integrate new sequences to the feature set, i.e., $F_\sigma = F_{\sigma-1} \cup S_\sigma$
   c. Increment the frequency threshold; i.e., $\sigma = \sigma + 1$
WHILE ($S_{\sigma-1}$ contain at least one sequence of two or more words not included in $F_{\sigma-2}$)
4. Build the training instances using the discovered Boolean features
5. Give the learning algorithm the training instances and perform training

**CLASSIFICATION**
1. Build the representation of $d$ in accordance to the training feature space
2. Let the trained classifier label the new instance
Maximal frequent sequences

- Identify authors of poems written by different Mexican poets

<table>
<thead>
<tr>
<th>Poets</th>
<th>Number of documents</th>
<th>Size of Vocabulary</th>
<th>Number of Phrases</th>
<th>Average Words by Documents</th>
<th>Average Phrases by Documents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efrain Huerta</td>
<td>48</td>
<td>3831</td>
<td>510</td>
<td>236.5</td>
<td>22.3</td>
</tr>
<tr>
<td>Jaime Sabines</td>
<td>80</td>
<td>3955</td>
<td>717</td>
<td>155.8</td>
<td>17.4</td>
</tr>
<tr>
<td>Octavio Paz</td>
<td>75</td>
<td>3335</td>
<td>448</td>
<td>162.6</td>
<td>27.2</td>
</tr>
<tr>
<td>Rosario Castellanos</td>
<td>80</td>
<td>4355</td>
<td>727</td>
<td>149.3</td>
<td>16.4</td>
</tr>
<tr>
<td>Ruben Bonifaz</td>
<td>70</td>
<td>4769</td>
<td>720</td>
<td>178.3</td>
<td>17.3</td>
</tr>
</tbody>
</table>

- Baseline results

- Maximal frequent sequences approach
Maximal frequent sequences

• MFS can discover interesting and useful patterns, however, extracting all of the MFS is a time consuming process

• MFS do not exploit information about the labels in training documents (it is an unsupervised method)

• Informativeness of patterns heavily depends on the frequency threshold $\sigma$
Phrase patterns

- A text is considered an ordered list of sentences, where each sentence is an unordered set of words.

- The goal is to identify interesting *sequences of sets of words*. The order is at the sentence level.

- Sequential patterns are extracted per each category.

Novel-representations text-classification
Representation-for-documents
Authorship-attribution
Return-a-effective-classification model
Phrase patterns

• Similar to MFS: Sequential patterns aim at discovering temporal relations between items (words) in a database (corpus)

• Main idea: extending work on mining association rules to extract meaningful sequential patterns

<table>
<thead>
<tr>
<th>Mining association rules</th>
<th>Text categorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client</td>
<td>Text</td>
</tr>
<tr>
<td>Item</td>
<td>Word</td>
</tr>
<tr>
<td>Items/transaction</td>
<td>Sentence (set of words)</td>
</tr>
<tr>
<td>Data</td>
<td>Position of the sentence in document</td>
</tr>
</tbody>
</table>
Phrase patterns

• Let $s = s_1, \ldots, s_k$ be a sequence, the support of $s$ is defined as:

$$sp(s) = \frac{\# \text{texts matching } s}{\# \text{texts}}$$

• Sequences with a support higher than $\text{minsup}$ are considered for the next step. Frequent patterns are used to generate rules of the form:

$$\gamma : <s_1, \ldots, s_k> \rightarrow C_i$$

• The confidence of a frequent pattern is defined as follows:

$$\text{conf}(\gamma) = \frac{\# \text{text from } C_i \text{ matching } <s_1, \ldots, s_k>}{\# \text{text matching } <s_1, \ldots, s_k>}$$

• Classification is done with a KNN scheme over rules with highest confidence
Phrase patterns

- Interesting patterns can be obtained with this formulation

- Class-information is considered in obtaining sequential rules

- Similar results to BOW using SVMs

- A large number of rules can be obtained and (as with MFS) extracting sequential patterns is a time consuming process
The locally weighted bag-of-words framework

• **LOWBOW**: an attempt to enrich BoW representations with sequential information *without defining/generating new terms/patters*

• Each document is represented by a set of local histograms computed across the whole document but smoothed by kernels and centered at different document locations

• LOWBOW-based document representations can preserve sequential information in documents
The locally weighted bag-of-words framework

- A document is a sequence of N words, it can be seen as a categorical time series:

\[ d_i = \langle d_{i,1}, \ldots, d_{i,N} \rangle \quad \text{with} \quad d_{i,j} \in V \]

- Idea: smooth temporarily this categorical times series with a Kernel: \( K_{\mu,\sigma}(x) \)

\[
K_{\mu,\sigma}(x) = \text{Beta}(x; \beta \frac{\mu}{\sigma}, \beta \frac{1-\mu}{\sigma})
\]

\[
K_{\mu,\sigma}(x) = \begin{cases} 
N(x; \mu, \sigma) & x \in [0,1] \\
0 & \text{otherwise}
\end{cases}
\]
The locally weighted bag-of-words framework

Let:

\[
\delta_c(d_i)(k, j) = \begin{cases} \frac{c}{1 + c|V|} & d_{i,k} \neq j \\ \frac{1 + c}{1 + c|V|} & d_{i,k} = j \end{cases}
\]

Denote the weight of term \( j \) at position \( k \) of document \( i \), for \( k \) a subset of locations at the documents.

The LOWBOW representation of the word sequence \( d_i \) is:

\[
Y(d_i) = \{ Y_\mu(d_i) : \mu \in [0,1] \}
\]

where \( Y_\mu(d_i) \) is the local word histogram at \( \mu \) defined by

\[
[Y_\mu(d_i)] = \int_0^1 \phi(\delta_c(d_i))(t, j) K\mu, \sigma(t) dt
\]
The locally weighted bag-of-words framework

Novel representations and methods in text classification
Manuel Montes-y-Gómez & Hugo Jair Escalante

Two core components of any classification system are the adopted representation for documents and the classification model itself. This tutorial deals with recent advances and developments on both components. The default representation for documents in text classification is the bag-of-words (BOW), where weighting schemes similar to those used in information retrieval are adopted. Whereas this representation has proven to be very helpful for thematic text classification, in novel, non-thematic text classification problems (e.g., authorship attribution, sentiment analysis and opinion mining, etc.), the standard BOW can be outperformed by other advanced representations.

This course is focused on three document representations that have proved to be useful for capturing more information than the raw occurrence of terms in documents as in BOW. The considered representations are: locally weighted BOW, distributional term representations, concise representations and graph-based representations. Likewise, the tutorial covers recent developments in the task of building classification models. Specifically, we consider contextual classification techniques and full model selection methods. The former approach is focused in the design of classifiers that consider the neighborhood of a document for making better predictions. The latter formulation focuses in the development of automatic methods for building classification systems, that is, black box tools that receive as input a data set and return a very effective classification model.
The locally weighted bag-of-words framework

Two core components of any classification system are the adopted representation for documents and the classification model itself. This tutorial deals with recent advances and developments on both components. The default representation for documents in text classification is the bag-of-words (BOW), where weighting schemes similar to those used in information retrieval are adopted. Where this representation has proven to be very helpful for thematic text classification, in novel, non-thematic text classification problems (e.g., authorship attribution, sentiment analysis and opinion mining, etc.), the standard BOW can be outperformed by other advanced representations.

This course is focused on three document representations that have proved to be useful for capturing more information than the raw occurrence of terms in documents as in BOW. The considered representations are: locally weighted BOW, distributional term representations, concise representations and graph-based representations. Likewise, the tutorial covers recent developments in the task of building classification models. Specifically, we consider contextual classification techniques and full model selection methods. The former approach is focused in the design of classifiers that consider the neighborhood of a document for making better predictions. The latter formulation focuses in the development of automatic methods for building classification systems, that is, black box tools that receive as input a data set and return a very effective classification model.

Weight the contribution of terms according to Gaussians at the different locations
The locally weighted bag-of-words framework

Novel representations and methods in text classification
Manuel Montes-y-Gómez & Hugo Jair Escalante

Two core components of any classification system are the adopted representation for documents and the classification model itself. The default representation for documents in text classification is the bag-of-words (BOW), where weighting schemes similar to those used in information retrieval are adopted. Whereas this representation has proven to be very helpful for thematic text classification, in novel, non-thematic text classification problems (e.g., authorship attribution, sentiment analysis and opinion mining, etc.), the standard BOW can be outperformed by other advanced representations.

This course is focused on three document representations that have proved to be useful for capturing more information than the raw occurrence of terms in documents as in BOW. The considered representations are: locally weighted BOW, distributional term representations, concise representations and graph-based representations. Likewise, the tutorial covers recent developments in the task of building classification models. Specifically, we consider contextual classification techniques and full model selection methods. The former approach is focused in the design of classifiers that consider the neighborhood of a document for making better predictions. The latter formulation focuses in the development of automatic methods for building classification systems, that is, black box tools that receive as input a data set and return a very effective classification model.

Weight the contribution of terms according to Gaussians at the different locations
Novel representations and methods in text classification
Manuel Montes-y-Gómez & Hugo Jair Escalante

Two core components of any classification system are the adopted representation for documents and the classification model itself. This tutorial deals with recent advances and developments on both components. The default representation for documents in text classification is the bag-of-words (BOW), where weighting schemes similar to those used in information retrieval are adopted. Whereas this representation has proven to be very helpful for thematic text classification, in novel, non-thematic text classification problems (e.g., authorship attribution, sentiment analysis and opinion mining, etc.), the standard BOW can be outperformed by other advanced representations.

This course is focused on three document representations that have proved to be useful for capturing more information than the raw occurrence of terms in documents as in BOW. The considered representations are: locally weighted BOW, distributional term representations, concise graph-based representations. Likewise, the tutorial covers recent developments in the task of building classification models. Specifically, we consider contextual classification techniques and full model selection methods. The former approach is focused in the design of classifiers that consider the neighborhood of a document for making better predictions. The latter formulation focuses in the development of automatic methods for building classification systems, that is, black box tools that receive as input a data set and return a very effective classification model.
Novel representations and methods in text classification
Manuel Montes-y-Gómez & Hugo Jair Escalante

Two core components of any classification system are the adopted representation for documents and the classification model itself. This tutorial deals with recent advances and developments on both components. The default representation for documents in text classification is the bag-of-words (BOW), where weighting schemes similar to those used in information retrieval are adopted. Whereas this representation has proven to be very helpful for thematic text classification, in novel, non-thematic text classification problems (e.g., authorship attribution, sentiment analysis and opinion mining, etc.), the standard BOW can be outperformed by other advanced representations.

This course is focused on three document representations that have proved to be useful for capturing more information than the raw occurrence of terms in documents as in BOW. The considered representations are: locally weighted BOW, distributional term representations, and concise graph-based representations. Likewise, the tutorial covers recent developments in the task of building classification models. Specifically, we consider contextual classification techniques and full model selection methods. The former approach is focused in the design of classifiers that consider the neighborhood of a document for making better predictions. The latter formulation focuses in the development of automatic methods for building classification systems, that is, black box tools that receive as input a data set and return a very effective classification model.

Weight the contribution of terms according to Gaussians at the different locations
LHs: position + frequency weighting

Position weighting

Kernel smoothing

Document

Kernel locations
The locally weighted bag-of-words framework

- A set of histograms, each weighted according to selected positions in the document

\[
d_i = \{dl_i^1, \ldots, dl_i^k\} \quad \text{and} \quad dl_i^j = d_i \times K_{\mu_j, \sigma}^S
\]
The locally weighted bag-of-words framework

- Standard bag-of-words:

\[ d_i = \left[ x_{i,1}, \ldots, x_{i,|V|} \right] \]
The locally weighted bag-of-words framework

- Documents represented under LOWBOW can be used for text categorization, using an appropriate distance measure (e.g.):

\[ D(\theta, \eta) = \arccos \left( \sum_{i=1}^{m} \sqrt{\theta_i \eta_i} \right) \quad \theta, \eta \in \mathbf{P}_{m+1} \]
The locally weighted bag-of-words framework

• Text segmentation:

• Taking the gradient norm of the lowbow curve: $\| Y_\mu (d_i) \|_2$
The locally weighted bag-of-words framework

- Text segmentation:

- PCA (left) and MDS (right) projections
LOWBOW for authorship attribution

• **Authorship attribution:** Given texts of uncertain authorship and texts from a set of candidate authors, the task is to map the uncertain texts onto their true authors among the candidates.

• Applications include: fraud detection, spam filtering, computer forensics and plagiarism detection
LOWBOW for authorship attribution

• LOWBOW acts as an expansion of the BOW approach that can be particularly suitable for AA

• Local histograms incorporate sequential information that reveal clues about the writing style of authors

• **Hypothesis:** Authors use similar distributions of certain words when writing documents

• We explore the use of LOWBOW for AA using character n-grams
LOWBOW for authorship attribution

Training texts of author A

\[ x_{A,1}, x_{A,2}, x_{A,3} \]

Classifier training

Training texts of author B

\[ x_{B,1}, x_{B,2} \]

Attribution model

Text of unknown authorship

\[ x_u \]

Most likely author
LOWBOW for authorship attribution

- How to take advantage of the multiple vectors associated to each document:
  - Combining the vectors (LOWBOW histogram)
    \[ L_i = \sum_{j=1}^{k} dl_i^j \]
  - Use the set of vectors to represent the document (BOLH)
    \[ L_i = \{ dl_i^1, ..., dl_i^k \} \]

- Classifier: Support vector machine
  \[ f(x) = \sum_i \alpha_i y_i K(x_i, x) - b \]
LOWBOW for authorship attribution

- Kernels for BOLHs

\[ K(P, Q) = e^{-\frac{1}{\gamma}D(P, Q)^2} \]

<table>
<thead>
<tr>
<th>Kernel</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diffusion</td>
<td>( D(P, Q) = \sum_{l=1}^{k} \arccos\left(\left&lt;\sqrt{p_l}, \sqrt{q_l}\right&gt;\right) )</td>
</tr>
<tr>
<td>Earth mover’s distance</td>
<td>( EMD(P, Q) )</td>
</tr>
<tr>
<td>Euclidean</td>
<td>( D(P, Q) = \sum_{l=1}^{k} \sum_{i=1}^{</td>
</tr>
<tr>
<td>Chi-squared</td>
<td>( D(P, Q) = \sum_{l=1}^{k} \sum_{i=1}^{</td>
</tr>
</tbody>
</table>
Experimental settings

• We consider a subset of RCV-I, documents written by 10 authors (about the same subject); 50 documents are available for training and 50 for testing for each author.

• Experiments using words and 3-grams at the character level were performed, different number of locations and scale parameters were evaluated, we report the settings that showed better performance.

• The 2500 most frequent terms were used to obtain the representations.
Experimental settings

- Three settings were considered:
  - **Balanced data set (BC):** 50 documents for training per author
  - **Reduced data set (RBC):** 4 subsets using 1, 3, 5 and 10 training documents per author
  - **Imbalanced data set (IRBC):** 3 subsets generated with a Gaussian distribution over authors using at least 2, 5, 10 and at most 10, 10, and 20 documents, respectively.
Balanced data set (BC)

BOW a strong baseline

<table>
<thead>
<tr>
<th>Method</th>
<th>Parameters</th>
<th>Words</th>
<th>Char. N-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>-</td>
<td>78.2%</td>
<td>75.0%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>k = 2; σ = 0.2</td>
<td>75.8%</td>
<td>72.0%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>k = 5; σ = 0.2</td>
<td>77.4%</td>
<td>75.2%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>k = 20; σ = 0.2</td>
<td>77.4%</td>
<td>75.0%</td>
</tr>
</tbody>
</table>

LOWBOW histograms

<table>
<thead>
<tr>
<th>k</th>
<th>Euc.</th>
<th>Diff.</th>
<th>EMD</th>
<th>Chi^2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Words</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>78.6%</td>
<td>81.0%</td>
<td>75.0%</td>
<td>75.4%</td>
</tr>
<tr>
<td>5</td>
<td>77.6%</td>
<td>82.0%</td>
<td>72.0%</td>
<td>77.2%</td>
</tr>
<tr>
<td>20</td>
<td>79.2%</td>
<td>80.8%</td>
<td>75.2%</td>
<td>79.0%</td>
</tr>
</tbody>
</table>

|     | Character N-grams |       |     |       |
| 2   | 83.4% | 82.8% | 84.4% | 83.8% |
| 5   | 83.4% | 84.2% | 82.2% | 84.6% |
| 20  | 84.6% | 86.4% | 81.0% | 85.2% |

BOLH

BOLHs obtained better performance
Balanced data set (BC)

BOW a strong baseline

<table>
<thead>
<tr>
<th>Method</th>
<th>Parameters</th>
<th>Words</th>
<th>Char. N-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>-</td>
<td>78.2%</td>
<td>75.0%</td>
</tr>
<tr>
<td>LOWBOW k = 2; σ = 0.2</td>
<td>75.8%</td>
<td>72.0%</td>
<td></td>
</tr>
<tr>
<td>LOWBOW k = 5; σ = 0.2</td>
<td>77.4%</td>
<td>75.2%</td>
<td></td>
</tr>
<tr>
<td>LOWBOW k = 20; σ = 0.2</td>
<td>77.4%</td>
<td>75.0%</td>
<td></td>
</tr>
</tbody>
</table>

LOWBOW histograms

<table>
<thead>
<tr>
<th>k</th>
<th>Euc.</th>
<th>Diff.</th>
<th>EMD</th>
<th>Chi²</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Words</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>78.6%</td>
<td>81.0%</td>
<td>75.0%</td>
<td>75.4%</td>
</tr>
<tr>
<td>5</td>
<td>77.6%</td>
<td>82.0%</td>
<td>72.0%</td>
<td>77.2%</td>
</tr>
<tr>
<td>20</td>
<td>79.2%</td>
<td>80.8%</td>
<td>75.2%</td>
<td>79.0%</td>
</tr>
</tbody>
</table>

Character N-grams

<table>
<thead>
<tr>
<th>k</th>
<th>Euc.</th>
<th>Diff.</th>
<th>EMD</th>
<th>Chi²</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>83.4%</td>
<td>82.8%</td>
<td>84.4%</td>
<td>83.8%</td>
</tr>
<tr>
<td>5</td>
<td>83.4%</td>
<td>84.2%</td>
<td>82.2%</td>
<td>84.6%</td>
</tr>
<tr>
<td>20</td>
<td>84.6%</td>
<td>86.4%</td>
<td>81.0%</td>
<td>85.2%</td>
</tr>
</tbody>
</table>

BOLH

BOLHs obtained better performance
Balanced data set (BC)

BOW a strong baseline

<table>
<thead>
<tr>
<th>Method</th>
<th>Parameters</th>
<th>Words</th>
<th>Char. N-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>-</td>
<td>78.2%</td>
<td>75.0%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>k = 2; σ = 0.2</td>
<td>75.8%</td>
<td>72.0%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>k = 5; σ = 0.2</td>
<td>77.4%</td>
<td>75.2%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>k = 20; σ = 0.2</td>
<td>77.4%</td>
<td>75.0%</td>
</tr>
</tbody>
</table>

LOWBOW histograms

<table>
<thead>
<tr>
<th>k</th>
<th>Euc.</th>
<th>Diff.</th>
<th>EMD</th>
<th>Chi²</th>
</tr>
</thead>
<tbody>
<tr>
<td>Words</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>78.6%</td>
<td>81.0%</td>
<td>75.0%</td>
<td>75.4%</td>
</tr>
<tr>
<td>5</td>
<td>77.6%</td>
<td>82.0%</td>
<td>72.0%</td>
<td>77.2%</td>
</tr>
<tr>
<td>20</td>
<td>79.2%</td>
<td>80.8%</td>
<td>75.2%</td>
<td>79.0%</td>
</tr>
</tbody>
</table>

Character N-grams

<table>
<thead>
<tr>
<th>k</th>
<th>Euc.</th>
<th>Diff.</th>
<th>EMD</th>
<th>Chi²</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>83.4%</td>
<td>82.8%</td>
<td>84.4%</td>
<td>83.8%</td>
</tr>
<tr>
<td>5</td>
<td>83.4%</td>
<td>84.2%</td>
<td>82.2%</td>
<td>84.6%</td>
</tr>
<tr>
<td>20</td>
<td>84.6%</td>
<td>86.4%</td>
<td>81.0%</td>
<td>85.2%</td>
</tr>
</tbody>
</table>

BOLH

BOLHs obtained better performance
Reduced balanced data sets

- Using words as terms

<table>
<thead>
<tr>
<th>Method \ dataset</th>
<th>1-doc</th>
<th>3-docs</th>
<th>5-docs</th>
<th>10-docs</th>
<th>50-docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>36.8%</td>
<td>57.1%</td>
<td>62.4%</td>
<td>69.9%</td>
<td>78.2%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>37.9%</td>
<td>55.6%</td>
<td>60.5%</td>
<td>69.3%</td>
<td>77.4%</td>
</tr>
<tr>
<td>Diff. Kernel</td>
<td>52.4%</td>
<td>63.3%</td>
<td>69.2%</td>
<td>72.8%</td>
<td>82.0%</td>
</tr>
<tr>
<td>Reference</td>
<td>-</td>
<td>-</td>
<td>53.4%</td>
<td>67.8%</td>
<td>80.8%</td>
</tr>
</tbody>
</table>

- Using character n-grams as terms

<table>
<thead>
<tr>
<th>Method \ dataset</th>
<th>1-doc</th>
<th>3-docs</th>
<th>5-docs</th>
<th>10-docs</th>
<th>50-docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>65.3%</td>
<td>71.9%</td>
<td>74.2%</td>
<td>76.2%</td>
<td>75.0%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>61.9%</td>
<td>71.6%</td>
<td>74.5%</td>
<td>73.8%</td>
<td>75.0%</td>
</tr>
<tr>
<td>Diff. Kernel</td>
<td>70.7%</td>
<td>78.3%</td>
<td>80.6%</td>
<td>82.2%</td>
<td>86.4%</td>
</tr>
<tr>
<td>Reference</td>
<td>-</td>
<td>-</td>
<td>53.4%</td>
<td>67.8%</td>
<td>80.8%</td>
</tr>
</tbody>
</table>
Imbalanced data sets

• Using words as terms

<table>
<thead>
<tr>
<th>Method \ dataset</th>
<th>2-10</th>
<th>5-10</th>
<th>10-20</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>62.3%</td>
<td>67.2%</td>
<td>71.2%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>61.1%</td>
<td>67.4%</td>
<td>71.5%</td>
</tr>
<tr>
<td>Diff. Kernel</td>
<td>66.6%</td>
<td>70.7%</td>
<td>74.1%</td>
</tr>
<tr>
<td>Reference</td>
<td>49.2%</td>
<td>59.8%</td>
<td>63.0%</td>
</tr>
</tbody>
</table>

• Using character n-grams as terms

<table>
<thead>
<tr>
<th>Method \ dataset</th>
<th>2-10</th>
<th>5-10</th>
<th>10-20</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>70.1%</td>
<td>73.4%</td>
<td>73.1%</td>
</tr>
<tr>
<td>LOWBOW</td>
<td>70.8%</td>
<td>72.8%</td>
<td>72.1%</td>
</tr>
<tr>
<td>Diff. Kernel</td>
<td>77.8%</td>
<td>80.5%</td>
<td>82.2%</td>
</tr>
<tr>
<td>Reference</td>
<td>49.2%</td>
<td>59.8%</td>
<td>63.0%</td>
</tr>
</tbody>
</table>
LOWBOW for authorship attribution

• Conclusions:
  – Sequential information encoded in local histograms is useful for AA. Character-level representations, which have proved to be very effective for AA can be further improved by adopting a local histogram formulation.

  – Our results are superior to state of the art approaches, with improvements ranging from 2%-6% in balanced data sets and from 14%-30% in imbalanced data sets (larger improvements were observed in challenging conditions).

  – In preliminary experiments with short texts we have found that LOWBOW does not work very well.
Research opportunities with LOWBOW

• Automatically-dynamically setting the number of local histograms for documents according to their length

• Studying the performance of local histograms in terms of length of documents, training set size, sparseness, narrowness of domain, etc.

• Profile-based authorship attribution using local histograms

• Learning the appropriate smoothing function from data
Discussion

• One of the main limitations of the BOW formulation is its inability to incorporate sequential information.

• Several extensions/alternatives to BOW have been proposed so far, each of which has limitations and advantages with respect to each other.

• Too much work to do in this topic = research opportunities.
References


Novel representations and methods in text classification

SYNTACTIC INFORMATION IN TEXT CLASSIFICATION
Outline

• Complex linguistic features for text classification
• Use of syntactic features in authorship attribution
  – Brief review
  – Syntactic-based n-grams as features
  – AA using Probabilistic Context-Free Grammars
• Final remarks
Background

• Long history on the use of complex linguistic features in information retrieval (refer to TREC reports)
  – Have been used: lemmas, POS information, named entities, noun phrases, complex nominals, syntactic tuples such as subject-verb, verb-object, etc.

• General conclusion: the high computational cost of the adopted NLP algorithms, the small improvement produced over simple BoW representation, and the lack of accurate WSD tools are the reasons for the failure of NLP in document retrieval
Linguistic features in text classification

• Are they useful for text classification?
  – IR and text classification are similar tasks, both are rely on thematic similarities.
  – Strong evidence indicates that POS information, complex nominals, and word senses are not adequate to improve TC accuracy

Useful for other textual-based classification tasks?

Features in authorship attribution

• AA deals with the definition of features that quantify the **writing style of authors**, and with the application of methods able to learn from that kind of features.

  – **Lexical features** $\rightarrow$ stylometric measures, words n-grams, function words
  
  – **Character-based features** $\rightarrow$ n-grams
  
  – **Syntactic features**
  
  – **Semantic features** $\rightarrow$ Use of synonyms and hyponyms, LSI
  
  – **Domain specific features** $\rightarrow$ Use/type of greetings, signatures, indentation, etc.

---

Syntactic features in AA

• The idea is that authors tend to use similar syntactic patterns unconsciously.
  – Strong authorial fingerprint

• Two basic approaches:
  – Use POS tag frequencies or POS n-gram frequencies as features
  – Apply a chunker, and use phrase counts as features

• Recent approaches:
  – Using syntactic-based n-grams as features
  – Using probabilistic context free grammars as language models for classification.
Syntactic n-grams

• Sn-grams are obtained based on the order in which the elements are presented in syntactic trees.
  – Constructed by following a path in the tree, rather than taking words as they appear in the text.

• Because sn-grams are based on syntactic relations of words, each word is bound to its real neighbors, ignoring the arbitrariness that is introduced by the surface structure

An example of sn-grams

- Common word n-grams:
  - eat with

- Common word sn-grams:
  - eat with, with spoon; eat with spoon

- Ignoring function words we would obtain:
  - eat spoon
Other variants of sn-grams

- In addition to word sn-grams, it is possible to build:
  - POS sn-grams
  - Sn-grams of syntactic relations tags (SR tags), where the elements are names of syntactic relations
  - Mixed sn-grams: composed by mixed elements like words (lexical units), POS tags and/or SR tags.

Sn-grams of SR tags

- \textit{nsubj} \rightarrow \textit{nn}
- \textit{dobj} \rightarrow \textit{amod}
- \textit{dobj} \rightarrow \textit{prep} \rightarrow \textit{pobj} \rightarrow \textit{amod}

\textit{Economic news have little effect on financial markets}
### Results

<table>
<thead>
<tr>
<th>Author</th>
<th>Training Novels</th>
<th>Size (MB)</th>
<th>Classification Novels</th>
<th>Size (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Booth Tarkington</td>
<td>8</td>
<td>3.6</td>
<td>5</td>
<td>1.8</td>
</tr>
<tr>
<td>George Vaizey</td>
<td>8</td>
<td>3.8</td>
<td>5</td>
<td>2.1</td>
</tr>
<tr>
<td>Louis Tracy</td>
<td>8</td>
<td>3.6</td>
<td>5</td>
<td>2.2</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>24</strong></td>
<td><strong>11</strong></td>
<td><strong>15</strong></td>
<td><strong>6.1</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Profile size</th>
<th>sn-grams of SR tags</th>
<th>n-grams of POS tags</th>
<th>Character based n-grams</th>
<th>Word based n-grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>100%</td>
<td>90%</td>
<td>76%</td>
<td>81%</td>
</tr>
<tr>
<td>1,000</td>
<td>100%</td>
<td>90%</td>
<td>86%</td>
<td>71%</td>
</tr>
<tr>
<td>4,000</td>
<td>100%</td>
<td>100%</td>
<td>95%</td>
<td>95%</td>
</tr>
<tr>
<td>7,000</td>
<td>100%</td>
<td>100%</td>
<td>90%</td>
<td>90%</td>
</tr>
<tr>
<td>11,000</td>
<td>100%</td>
<td>95%</td>
<td>100%</td>
<td>90%</td>
</tr>
</tbody>
</table>

- Profile size indicates the first most frequent n-grams/snigrams
AA using Probabilistic Context Free Grammars

• **Idea**: use of syntactic information by building complete models of each author’s syntax to distinguish between authors.

• **How**: build a probabilistic context free grammar (PCFG) for each author and use this grammar as a language model for classification.
  
  – A PCFG is a probabilistic version of a CFG where each production has a probability
  
  – Probability of a sentence/derivation is the product of the probabilities of its productions

General procedure

**Input** – A training set of documents labeled with author names and a test set of documents with unknown authors.

1. Train a statistical parser on a generic corpus like the WSJ or Brown corpus.

2. Treebank each training document using the parser trained in Step 1.

3. Train a PCFG $G_i$ for each author $A_i$ using the treebanked documents for that author.

4. For each test document, compute its likelihood for each grammar $G_i$ by multiplying the probability of the top PCFG parse for each sentence.

5. For each test document, find the author $A_i$ whose grammar $G_i$ results in the highest likelihood score.

**Output** – A label (author name) for each document in the test set.

- Generate a **parse tree** for each training document
- Estimate a **grammar** and its parameters from the assembled “tree-bank”
- Compute **probabilities** for each document, for each grammar
- **Select the author** (grammar) with the highest probability
### Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th># authors</th>
<th># words/auth</th>
<th># docs/auth</th>
<th># sent/auth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Football</td>
<td>3</td>
<td>14374.67</td>
<td>17.3</td>
<td>786.3</td>
</tr>
<tr>
<td>Business</td>
<td>6</td>
<td>11215.5</td>
<td>14.16</td>
<td>543.6</td>
</tr>
<tr>
<td>Travel</td>
<td>4</td>
<td>23765.75</td>
<td>28</td>
<td>1086</td>
</tr>
<tr>
<td>Cricket</td>
<td>4</td>
<td>23357.25</td>
<td>24.5</td>
<td>1189.5</td>
</tr>
<tr>
<td>Poetry</td>
<td>6</td>
<td>7261.83</td>
<td>24.16</td>
<td>329</td>
</tr>
</tbody>
</table>

#### Words

<table>
<thead>
<tr>
<th>Dataset</th>
<th>MaxEnt</th>
<th>Naive Bayes</th>
<th>Bigram-I</th>
<th>PCFG</th>
<th>PCFG-I</th>
<th>PCFG-E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Football</td>
<td>84.45</td>
<td>86.67</td>
<td>86.67</td>
<td>93.34</td>
<td>80</td>
<td>91.11</td>
</tr>
<tr>
<td>Business</td>
<td>83.34</td>
<td>77.78</td>
<td>90.00</td>
<td>77.78</td>
<td>85.56</td>
<td>91.11</td>
</tr>
<tr>
<td>Travel</td>
<td>83.34</td>
<td>83.34</td>
<td>91.67</td>
<td>81.67</td>
<td>86.67</td>
<td>91.67</td>
</tr>
<tr>
<td>Cricket</td>
<td>91.67</td>
<td>95.00</td>
<td>91.67</td>
<td>86.67</td>
<td>91.67</td>
<td>95.00</td>
</tr>
<tr>
<td>Poetry</td>
<td>56.36</td>
<td>78.18</td>
<td>70.90</td>
<td>78.18</td>
<td>83.63</td>
<td>87.27</td>
</tr>
</tbody>
</table>

- **PCFG-I**: augments the training data with section of the Brown corpus; replicates the original data 3-4 times
- **PCFG-E**: an ensemble of MaxEnt, Bigram-I and PCFG-I
Final remarks

• Syntactic information is an important authorial fingerprint

• But, both syntactic and lexical information are useful in effectively capturing authors’ overall writing style
  – Mixed sn-grams are a good compromise between these two sources of information

• Some disadvantages of using syntactic-based features:
  – Syntactic parsing is required!
    • Can take considerable time
    • Problem of availability of parsers for some languages
  – Language-dependent procedure