Maschinelles Lernen – Theorie und Anwendung

Text Mining

Prof. Klaus-Robert Müller
Ulf Brefeld

Arbeitsgruppe Maschinelles Lernen

Why Text Mining?

- Textual information is ubiquitous.
  - WWW, news archives, linked document archives, ...
- Information extraction.
  - Relation and event extraction.
  - Find entities like names, date, time, location, ...
- Information retrieval.
  - Web search.
  - Find related (news) articles.
- Applications based on text mining:
  - Search engines (e.g., Yahoo, Google).
  - Recommender systems (e.g., Amazon).
  - Machine translation (e.g., babelfish).

Overview

Characteristics of Natural Language

Eigenschaften natürlicher Sprache

- Unendlich viele Ausdrücke.
- Rekursion:
  - *Der Bezug des Bettes des Hotels des Ermittlungsteams der Ursache des Absturzes des Systems ...*
  - *Systemabsturzursachenermittlungsteamhotelbettbezug*
- Konjunktion (Aufzählung):
  - *Am Sonntag fraß Sie sich durch einen Äpfel, zwei Bananen, drei Tomaten, vier Gurken, fünf Schokohasen, sechs ...*
- Hinzunahme neuer Basiselemente:
  - Entlehnung: *to go, Email, ...*
  - Kreativität: *unkaputtbar, Handy, ...*

Eigenschaften natürlicher Sprache (forts.)

- Synonyme: *zwölf*, *12* und *XIII* ; *Orange* und *Apfelsine*,...
- Homonyme: *Schloss* (Gebäude und Türschloss)
- Ambiguität: *Ich sehe den Mann mit dem Fernrohr*, *Staubecken*....

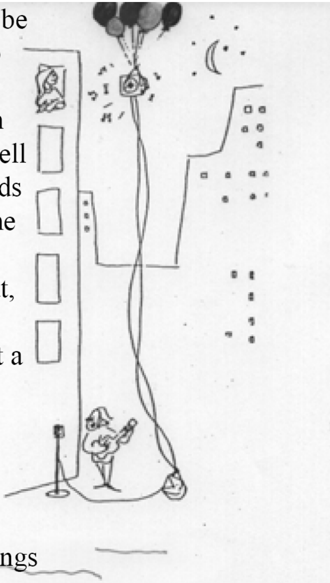
Desambiguierung

- Kontextabhängig.
- Beispiel: *Nach 14 Jahren Kohl, ...*
  - *... wollten wir mal wieder etwas anderes essen.*
  - *... lag die Arbeitslosigkeit bei x%*.
- Manchmal reicht das nicht...

Do you understand English?

If the balloons popped, the sound wouldn't be able to carry since everything would be too far away from the correct floor. A closed window would also prevent the sound from carrying, since most buildings tend to be well insulated. Since the whole operation depends on a steady flow of electricity, a break in the middle of the wire would also cause problems. Of course, the fellow could shout, but the human voice is not loud enough to carry that far. An additional problem is that a string could break on the instrument. Then there could be no accompaniment to the message. It is clear that the best situation would involve less distance. Then there would be fewer potential problems. With face to face contact, the least number of things could go wrong.
(Bransford and Johnson (1973))

If the balloons popped, the sound wouldn't be able to carry since everything would be too far away from the correct floor. A closed window would also prevent the sound from carrying, since most buildings tend to be well insulated. Since the whole operation depends on a steady flow of electricity, a break in the middle of the wire would also cause problems. Of course, the fellow could shout, but the human voice is not loud enough to carry that far. An additional problem is that a string could break on the instrument. Then there could be no accompaniment to the message. It is clear that the best situation would involve less distance. Then there would be fewer potential problems. With face to face contact, the least number of things could go wrong.



Common words in *Tom Sawyer*

word	frequency
the	3332
and	2972
a	1775
to	1725
of	1440
was	1161
it	1027
in	906
that	877
he	877
I	783
his	772
you	686
Tom	679

word frequency	freq. of frequency
1	3993
2	1292
3	664
4	410
5	243
6	199
7	172
8	131
9	82
10	91
11-50	540
51-100	99
> 100	102

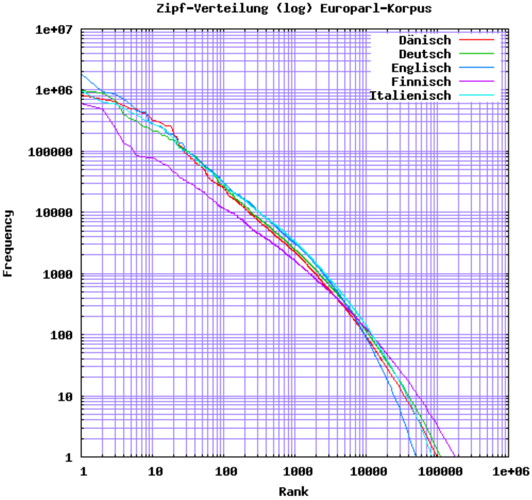
Zipf's Law

- Explores the relationship between the frequency of a word f and its rank r (i.e., its position in the list).

$$f \propto \frac{1}{r}$$

  or in other words: There is a constant k such that $f \cdot r = k$.
- Example: the 50$^{\text{th}}$ most common word should occur with three times the frequency of the 150$^{\text{th}}$ most common word.
- Zipf distribution: A few very frequent words, a middling number of medium frequency words, and many uncommon words.

Exemplary Zipf Distribution



Empirical evaluation of Zipf's law on *Tom Sawyer*

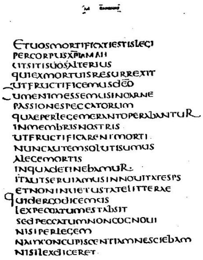
word	freq.	rank	$f \cdot r$
the	3332	1	3332
and	2972	2	5944
a	1775	3	5235
he	877	10	8770
but	410	20	8400
be	294	30	8820
there	222	40	8880
one	172	50	8600
about	158	60	9480
more	138	70	9660
never	124	80	9920
Oh	116	90	10440
two	104	100	10400

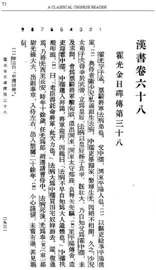
word	freq.	rank	$f \cdot r$
turned	51	200	10200
you'll	30	300	9000
name	21	400	8400
comes	16	500	8000
group	13	600	7800
lead	11	700	7700
friends	10	800	8000
begin	9	900	8100
family	8	1000	8000
brushed	4	2000	8000
sins	2	3000	6000
Could	2	4000	8000
Applausive	1	8000	8000

Document Representations

Tokenization

- Document = sequence of characters or symbols.
- Tokenization: Convert a document into a sequence of *tokens*.
- A token is a categorized block of text.
  - *Bello chases the cat.* → ⟨ Bello | chases | the | cat ⟩
- Frequently, prior knowledge is necessary:

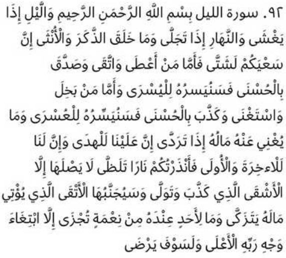




A CLASSICAL CHINESE READER

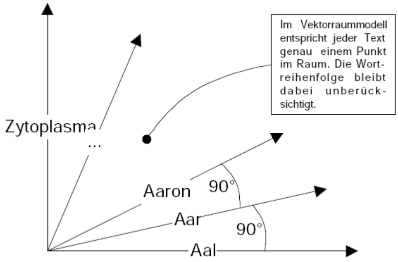
漢書卷六十八

霍光金日磾傳第三十八



٩٢. سورة الليل بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ وَاللَّيْلِ إِذَا
يَغْشَى وَالنَّهَارِ إِذَا تَجَلَّى وَمَا خَلَقَ الذَّكَرَ وَالْأُنْثَى إِنَّ
سَعْيَكُمْ لَشَتَّى فَأَمَّا مَنْ أَعْطَى وَاتَّقَى وَصَدَّقَ
بِالْحُسْنَى فَسَنُيَسِّرُهُ لِلْيُسْرَى وَأَمَّا مَنْ بَخِلَ
وَاسْتَغْنَى وَكَذَّبَ بِالْحُسْنَى فَسَنُيَسِّرُهُ لِلْعُسْرَى وَمَا
يُغْنِي عَنْهُ مَالُهُ إِذَا تَرَدَّى إِنَّ عَلَيْنَا لَلْهُدَى وَإِنَّ لَنَا
لَلْآخِرَةَ وَالْأُولَى فَأَنْذَرْتُكُمْ نَارًا تَلَظَّى لَا يَصْلَاهَا إِلَّا
الْأَشْقَى الَّذِي كَذَّبَ وَتَوَلَّى وَسَيُجَنَّبُهَا الْأَتْقَى الَّذِي يُؤْتِي
مَالَهُ يَتَزَكَّى وَمَا لِأَحَدٍ عِنْدَهُ مِنْ نِعْمَةٍ تُجْزَى إِلَّا ابْتِغَاءَ
وَجْهِ رَبِّهِ الْأَعْلَى وَلَسَوْفَ يَرْضَى

The Vector Space Model



- Documents are represented in a high-dimensional vector space.
- Axes are identified with tokens.
- Ordering of tokens is lost.
- Examples: Bag-of-words, TF.IDF representations.

Bag-of-Words Representation

- Let $D = \{d_1, \ldots, d_m\}$ be a set of documents.
- Build dictionary $\mathcal{D} = \bigcup_{d \in D} \{w : \text{token } w \text{ occurs in document } d\}$.
- Indicator function $[\![z]\!] = 1$ if z is true and 0 otherwise.

$$BOW(d_j) = \begin{pmatrix} [\![w_1 \in d_j]\!] \\ [\![w_2 \in d_j]\!] \\ \vdots \\ [\![w_{|\mathcal{D}|} \in d_j]\!] \end{pmatrix}$$

- Drawback: All tokens are equally important.

TF.IDF

- Term frequency: number of occurrences of term w_i in a document.
- Problem 1: Long documents have large term frequencies
  $\Rightarrow$ difficult for similarity measure.
- Solution: normalize term frequency.

$$TF(w_i) = \frac{TF(w_i)}{\sum_i TF(w_i)}$$

- Problem 2: Several words are irrelevant (e.g., *the*, *and*, ...)
- Solution: inverse document frequency.

$$IDF(w_i) = \frac{\#\ \text{documents}}{\#\ \text{documents containing}\ w_i}.$$

TF.IDF Representation

- TF.IDF representation of word w_i determined by $TF(w_i) \cdot IDF(w_i)$.
- TF.IDF representation of document d_j is given by

$$TF.IDF(d_j) = \begin{pmatrix} TF(w_1) \cdot IDF(w_1) \\ TF(w_2) \cdot IDF(w_2) \\ \vdots \\ TF(w_{|\mathcal{D}|}) \cdot IDF(w_{|\mathcal{D}|}) \end{pmatrix}$$

N-grams

- Ordering in BOW and TF.IDF representation is lost.
- BUT: neighboring tokens are not independent!
- N-grams represent sequences up to n tokens:

$$P(w_t|w_{t-n+1},\ldots,w_{t-1}) = \frac{P(w_{t-n+1},\ldots,w_t)}{P(w_{t-n+1},\ldots,w_{t-1})}$$

- Several n-gram representations are possible:
  - Occurrence: $NG(w_1,\ldots,w_n;d_j) = [\![(w_1,\ldots,w_n) \in d_j]\!]$
  - Frequency: $NG(w_1,\ldots,w_n;d_j) = \#((w_1,\ldots,w_n) \in d_j)$
  - Probabilistic: $NG(w_1,\ldots,w_n;d_j) = P(w_n|w_1,\ldots,w_n;d_j)$

N-gram Representations

- N-gram vector space has one dimension per n-gram.
- Let $\mathcal{N}$ consist of all possible $(|\mathcal{D}|^n)$ n-grams.
- The n-gram representation of document d_j is given by

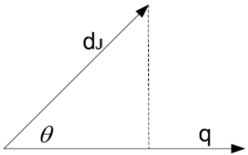
$$NGram(d_j) = \begin{pmatrix} NG(\mathbf{w}_1, d_j) \\ NG(\mathbf{w}_2, d_j) \\ \vdots \\ NG(\mathbf{w}_{|\mathcal{N}|}, d_j) \end{pmatrix}, \quad \mathbf{w} \in \mathcal{N}$$

- Parameter n needs to be chosen appropriately.

Normalization

- Problem: long texts result in long feature vectors.
  - Example: web search where queries hardly consist of more than 3 tokens.
- Solution: normalize feature vectors such that $\|\phi(d_j)\| = 1$ for all j.
- Similarity between document d_j and query q given by

$$sim(d_j, q) = \cos(d_j, q) = \frac{\langle d_j, q \rangle}{\|d_j\| \|q\|}$$



Dimensionality Reduction

- BOW, TF.IDF, and *n*-gram feature spaces are high-dimensional.
- Problems when using non-sparse learners (e.g., naïve Bayes).
- Stemming.
  - Strip off *affixes* (remove inflectional endings of words).
  - E.g., map for occurrences of *go*, *gone*, *going*, etc. to their root *go*.
  - Stemmers are freely available for many languages.
- Latent semantic indexing.
  - Similar to principal component analysis.
  - Map instances into new coordinate system.
  - New coordinates correspond to semantic concepts.
  - Reduce dimensionality by neglecting coordinates with low variance ($=$ hardly occurring semantic concepts).

Latent Semantic Indexing

	d_1	d_2	d_3	d_4	d_5	d_6
cosmonaut	1	0	1	0	0	0
astronaut	0	1	0	0	0	0
moon	1	1	0	0	0	0
car	1	0	0	1	1	0
truck	0	0	0	1	0	1

- Term-document matrix A.
- Find matrices T, S, and D such that $A = T \times S \times D^\mathsf{T}$.
  1. Compute eigenvalues $e_1, \ldots, e_p$ of $A^\mathsf{T} A$.
  2. Compute matrix D comprising the corresponding eigenvectors.
  3. Define $S = diag(e_1, \ldots, e_p)$.
  4. Compute T, for instance by Gram-Schmidt orthogonalization.

Applications

Classification of Text Documents

- Annotate text documents with class labels.
  - Binary classification.
  - Multi-class classification.
  - Multi-label classification.
- Applications:
  - Detect spam messages (binary).
  - Classify web pages into web directories (multi-class).
  - Classify news articles (multi-label).
- Learn a classifier from labeled documents.
  - For text linear classifiers have been proven to perform well.
  - E.g., linear support vector machines.

Support Vector Machines

- Binary text classification (e.g., ham vs. spam).
- SVMs minimize upper bound on regularized empirical risk.
- Labeled documents $\{(d_i, y_i)\}_{i=1}^m$ with $y_i \in \{+1, -1\}$.

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_m \\ \text{s.t.} \quad & \forall_{i=1}^m : \quad y_i(\langle w, \phi(d_i)\rangle + b) \geq 1 - \xi_i \\ & \forall_{i=1}^m : \quad \xi_i \geq 0. \end{aligned}$$

- Document representation $\phi(d)$.
- Easily generalized to multi-class and multi-label problems.
  - Strategy: one-against-one, one-against-all.

Evaluation of Text Classifiers

- Misclassification error rates not appropriate when $P(+1)$ small.
  - E.g., how good is an error of 5% when $P(+1) = 3\%$?
- Solution: measure performance of decision function $f(x)$
- Precision/Recall

$$
\begin{aligned}
Precision(f) = P(y = +1 | f(x) = +1) &= \frac{TP}{TP + FP}. \\
Recall(f) = P(f(x) = +1 | y = +1) &= \frac{TP}{TP + FN}.
\end{aligned}
$$

  - Breakeven point: $Prec(f) = Rec(f)$, F-measure: $F = \frac{2 \cdot Prec(f) \cdot Rec(f)}{Prec(f) + Rec(f)}$.
- Receiver Operating Characteristic (ROC).
  - Area under the ROC curve: $AUC(f) = P(f(x_{pos}) > f(x_{neg}))$.

Experiment

- Reuters-21578 data set (ModApte compilation).
  - News articles from Reuters news archive.
- 9603 training documents, 3299 test documents, 90 classes.
- Preprocessing: 9962 distinct terms in dictionary.
- Features: normalized term frequencies.
- Baselines: naïve Bayes, C4.5, Rocchio, *k*-nearest neighbors.

E.D. And F. MAN TO BUY INTO HONG KONG FIRM

The U.K. Based commodity house E.D. And F. Man Ltd and Singapore's Yeo Hiap Seng Ltd jointly announced that Man will buy a substantial stake in Yeo's 71.1 pct held unit, Yeo Hiap Seng Enterprises Ltd. Man will develop the locally listed soft drinks manufacturer into a securities and commodities brokerage arm and will rename the firm Man Pacific (Holdings) Ltd.

Empirical Results

	Bayes	Rocchio	C4.5	k-NN	SVM (poly) degree $d =$ 1	2	3	4	5	SVM (rbf) width $\gamma =$ 0.6	0.8	1.0	1.2
earn	95.9	96.1	96.1	97.3	98.2	98.4	**98.5**	98.4	98.3	**98.5**	98.5	98.4	98.3
acq	91.5	92.1	85.3	92.0	92.6	94.6	**95.2**	95.2	95.3	95.0	95.3	95.3	**95.4**
money-fx	62.9	67.6	69.4	78.2	66.9	72.5	75.4	74.9	**76.2**	74.0	75.4	**76.3**	75.9
grain	72.5	79.5	89.1	82.2	91.3	93.1	**92.4**	91.3	89.9	**93.1**	91.9	91.9	90.6
crude	81.0	81.5	75.5	85.7	86.0	87.3	88.6	**88.9**	87.8	**88.9**	89.0	88.9	88.2
trade	50.0	77.4	59.2	77.4	69.2	75.5	76.6	77.3	**77.1**	76.9	78.0	**77.8**	76.8
interest	58.0	72.5	49.1	74.0	69.8	63.3	67.9	73.1	**76.2**	74.4	75.0	**76.2**	76.1
ship	78.7	83.1	80.9	79.2	82.0	85.4	86.0	**86.5**	86.0	**85.4**	86.5	87.6	87.1
wheat	60.6	79.4	85.5	76.6	83.1	84.5	85.2	**85.9**	83.8	**85.2**	85.9	85.9	85.9
corn	47.3	62.2	87.7	77.9	86.0	86.5	85.3	**85.7**	83.9	**85.1**	85.7	85.7	84.5
microavg.	**72.0**	**79.9**	**79.4**	**82.3**	84.2	85.1	85.9	86.2	85.9	86.4	86.5	86.3	86.2
					combined: **86.0**					combined: **86.4**			

Fig. 2. Precision/recall-breakeven point on the ten most frequent Reuters categories and microaveraged performance over all Reuters categories. k-NN, Rocchio, and C4.5 achieve highest performance at 1000 features (with $k = 30$ for k-NN and $\beta = 1.0$ for Rocchio). Naive Bayes performs best using all features.

- SVMs well suited for sparse, high-dimensional feature spaces. (Joachims, 1998)

Summary & Further Applications

Summary

- Characteristics of natural language.
  - Infinitely many terms.
  - Ambiguous.
  - Disambiguation by context information.
- Document representations:
  - Bag-of-words, TF.IDF, n-grams.
  - Relevant for classification, clustering, and ranking tasks.
  - Dimensionality reduction techniques.
- Exemplary application.
  - Text classification with SVMs.
  - Performance measures.

Further Applications

- Potentially more challenging high-level tasks:
  - Natural language parsing.
  - Named entity recognition.
  - Named entity resolution.
  - Machine translation.
  - Sentiment prediction.
  - Document summarization.
  - Question answering.
  - ...

Literatur/Referenzen

- R. Baeza-Yates & B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley, 1999.
- X. Huang, A. Acero & H.-W. Hon, *Spoken Language Processing*, Prentice Hall, 2001.
- T. Joachims, Text Categorization with Support Vector Machines: Learning with Many Relevant Features, *Proceedings of the European Conference on Machine Learning (ECML)*, 1998.
- T. Joachims, A statistical learning model of text classification for support vector machines, *Proceedings of the Conference on Research and Development in Information Retrieval (SIGIR)*, 2001.
- T. Joachims, *Learning to Classify Text with Support Vector Machines*, Kluwer Academic Publishers / Springer, 2002.
- C. D. Manning & H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, 2003.