METODE INTELIGENTE DE REZOLVARE A PROBLEMELOR REALE

Laura Dioşan
Text mining

Facultatea de Matematică şi Informatică
Universitatea Babeş-Bolyai

1

1

- Natural Language Processing
  - Text classification
  - Language modelling
  - Machine translation
  - …

2

From Languages to Information

- Aim
  - Automatically extracting meaning and structure from:
    - Human language text and speech (news, social media, etc.)
    - Social networks
    - Genome sequences
  - Interacting with humans via language
    - Dialog systems/Chatbots
    - Question Answering
    - Recommendation Systems

3

From Languages to Information

- How?
  - Extracting information from language
    - Information Retrieval
    - Text Classification
    - Extracting Sentiment and Social Meaning
    - …
  - Interacting with humans via language
    - IBM's Watson
    - …
    - Counseling conversations
      - https://www.aclweb.org/anthology/Q16-1033.pdf
      - https://web.stanford.edu/class/cs124/lec/counseling_slides.pdf
    - Understanding police officer respect
      - https://www.pnas.org/content/114/25/6521
      - https://nlp.stanford.edu/robvoigt/124_lecture/

4

Text classification

- Definition
  - Text categorization
    - Assign (predefined) labels (categories) to some documents
    - Documents
      - Technical reports, web pages, messages, books, etc.
    - Categories:
      - Topics (art, economy)
      - Attributes (spams, ham)
- Data and various text classification problems
  - https://nlpprogress.com/english/text_classification.html

	Cuvinte	Documente
Învăţare supervizată	Etichetarea părţilor de vorbire	Clasificarea textelor, Filtrarea, Detectarea subiectelor
Învăţare nesupervizată	Indexarea semantică, construcţia automată a tezaurelor, extragerea cuvintelor cheie	Clusterizarea documentelor, Detectarea subiectelor

5

Text classification

- Automatic approaches
  - Based on knowledges
    - From experts
    - Encoded as rules
  - Based on learning
    - Experts label some training examples
    - The algorithm label the test examples
    - Learning
      - Supervised
      - Unsupervised
- Evaluation
  - Accuracy, Precision, Recall, AUC, etc.

6

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification

- Process
  - Data training analysis
    - **Document indexing**
      - Construct a document representation
        - Attributes / features
        - Weights of these features
      - Reduce the dimension of these representations
        - Feature selection
        - Feature extraction
    - Learning a classification model
  - Classification of test data
    - Text indexing
    - Apply the learnt model -> categories

7

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification

- Document indexing
  - Construct a document representation
    - Attributes / features
    - Weights of these features
  - 4 steps:
    1. Document linearization
    2. Filtering
    3. Canonical form
    4. Weighting

$\}$ Reduce the vocabulary size

8

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

Step 1: linearization (segmentation)
- Reduce the document to a vector of terms (attributes)
  - *bag of words model*
  - A matrix
    - Documents -> lines
    - Terms -> colomns
    - Each cell 1/0 – if the current term belongs to the current document
- 2-stage process for term identification
  - Format removal
    - E.g. Elimination of HTML tags
  - *Tokenization*
    - Parsing (segmentation)
    - Removal of punctuation mark
    - Conversion of capital letters

Iniţial	Liniarizat
Interactive query expansion modifies queries using terms from a user. Automatic query expansion expands queries automatically.	interactive query expansion modifies queries using terms from a user automatic query expansion expands queries automatically

9

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification -> indexing

Paul 2: filtering
- Select some terms that are able to
  - Describe the content of the document
  - Make a difference between two documents
- Removal of stopwords
  - From a predefined list
  - Based on their frequencies (under a given threshold)

Segmentat	Filtrat
interactive query expansion modifies queries using terms from a user automatic query expansion expands queries automatically	interactive query expansion modifies queries terms automatic query expansion expands queries automatically

10

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

- Step 3 – canonical form (normalization/standardisation)
  - Lematisation
    - Morphological analysis
    - Based on context
    - Ex. "better" → "good"
  - *Stemming*
    - Term-level
    - Ex. "computer", "computing", "compute" → "comput"
    - *stemming* algorithm
      - Martin Porter
      - WordNet

Filtrat	Redus
interactive query expansion modifies queries terms automatic query expansion expands queries automatically	interact queri expan modifi queri term automat queri expan expand queri automat

11

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

Step 4: weighting

- Pre-determined (manual)
  - Ex. Bag of Words
- Automatic (learnt)
  - Ex. embedded representations

12

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification - indexing

- Step 4: manual weighting
  - Usage of a particular model
  - Weights relative to
    - A single document
      - *term frequency – TF*
    - A collection of documents
      - *inverse document frequency – IDF*
    - hybrid *TF* and *IDF*
      - *TF* → cu cât un termen este mai frecvent într-un document, cu atât el este mai important pentru acel document
      - *IDF* → cu cât un termen apare în mai multe documente, cu atât el este mai puţin important în descrierea semanticii acelui document
  - Range of frequencies
    - Binary → presence / absence of a term (One-Hot encoding)
    - Real ([0,1]) → importance of that term
  - For a set of D documents and a set of T *termns, the weight* p_{ij} of term t_i in document d_j (i=1,2,...,|T|, j=1,2,...,|D|) can be:
    - binary: p_{ij} = 1, if t_i belongs to d_j
      0, otherwise
    - *TF*: $p_{ij}=tf_{ij}$ (no of appearances of term t_i in document d_j)
    - *TF.IDF*: $p_{ij}=tf_{ij}*\log_2(|D|/df_i)$, where df_i=no of documents where term t_i is found

13

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

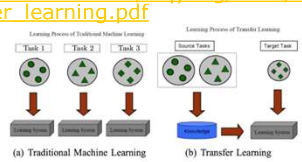
- Step 4 – weight learning
  - Unsupervised learnt representation
  - Classical models
    https://web.stanford.edu/class/cs124/lec/languagemodeling.pdf
  - https://web.stanford.edu/class/cs124/lec/lm2021.pdf
  -
    - Probability of a target word based on k previous words
    - Document-level context
      - Semantic relation (boat – water)
    - Count-based models
      - Latent Semantic Analysis (LSA)
      - Latent Dirichlet Allocation (LDA)
  - Modern models
    - Word-level context
      - Semantic similarity (boat – sheep)

14

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

- Step 4 – weight learning (word representations)
  - Methodologies
    - Traditional Machine Learning
    - Transfer learning
      https://www.cse.ust.hk/~qyang/Docs/2009/tkde_transfer_learning.pdf



15

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

Learning word representations (vectors)

- Why?
  - Embeddings = parameters -> they can be learnt
  - Share representation across tasks
  - Lower dimensional space
- How?
  - Pre-NN
    - 1990 - Latent Semantic Analysis (LSA) http://lsa.colorado.edu/papers/JASIS.lsi.90.pdf
    - 1992 – n-gram models https://www.aclweb.org/anthology/J92-4003.pdf
    - 2003 - Latent Dirichlet Allocation (LDA)—Documents are mixtures of topics and topics are mixtures of words https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf
  - NN-based
    - Word-level (2003 - ...)
    - Sentence (document) level (2014 - ...)
    - Contextual word-vectors (Word vectors compress all contexts into a *single vector) (2016 - ...)*

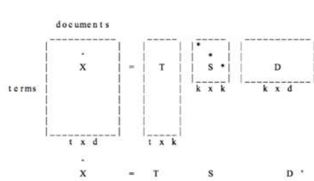
16

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

Learning word representations (vectors)

- How?
  - Pre-NN
    - 1990 - Latent Semantic Analysis (LSA)
      http://lsa.colorado.edu/papers/JASIS.lsi.90.pdf



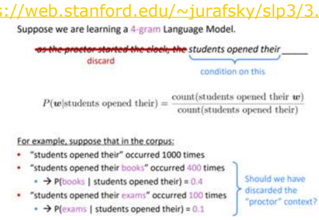
17

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

Learning word representations (vectors)

- How?
  - Pre-NN
    - 1992 – n-gram models
      https://www.aclweb.org/anthology/J92-4003.pdf
    - https://web.stanford.edu/~jurafsky/slp3/3.pdf



18

Text classification – indexing

Learning word representations (vectors)
- How? -> NN-based
  - Word-level (2003 - …)
    - 2003 - N-gram Neural language model (Montreal - Bengio)
      - https://www.jmlr.org/papers/volume3/bengio03a/bengio03a.pdf
      - Probability of the target word based on previous k words
      - Estimation of the probability by an ANN -> prediction of the next word in a sequence
        - Embedding layer -> word embeddings
        - Intermediate layer(s) -> intermediate representation (non-linear); standard (HardTanh layer) or recurrent layers
        - softmax layer for producing probabilities over vocabulary) -> main bottleneck
        - Cross-entropy loss (max the probability of the next word)

19

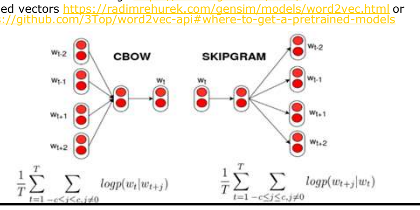
Text classification – indexing

Learning word representations (vectors)
- How? -> NN-based
  - Word-level (2003 - …)
    - 2008 – multi-task model (Princeton – Collobert)
      - https://ronan.collobert.com/pub/matos/2008_nlp_icml.pdf
      - Probability of MORE target words (a sequence of words)
      - Estimation of the probability by an ANN –similar to Bengio's model, but
        - a pairwise ranking criterion
        - outputs a higher score for a correct word sequence than for an incorrect one

20

Text classification – indexing

Learning word representations (vectors)
- How? -> NN-based
  - Word-level (2003 - …)
    - 2013 – word2vec (Google – Mikolov)
      - https://papers.nips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf
      - Continous BOW model -> try to predict a word based on its context (neighbours); input = neighbour words, output = target word
      - Skip-gram model -> try to predict context (neighbours) of a word; input = target word; output = neighbour words;
      - Visualisation of embeddings https://ronxin.github.io/wevi/
      - Trained vectors https://radimrehurek.com/gensim/models/word2vec.html or https://github.com/3Top/word2vec-api#where-to-get-a-pretrained-models



21

Text classification – indexing

Learning word representations (vectors)
- How? -> NN-based
  - Word-level (2003 - …)
    - 2014 – GloVe (Stanford – Pennington, Manning)
      - https://nlp.stanford.edu/projects/glove/
    - 2017 - fastText (Facebook – Mikolov)
      - https://arxiv.org/abs/1607.04606
      - https://radimrehurek.com/gensim/models/fasttext.html

22

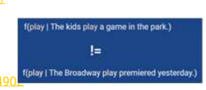
Text classification – indexing

Learning word representations (vectors)
- How? -> NN-based
  - Word-level (2003 - …)
  - Sentence (document) level (2014 - …)
    - 2014 - Paragraph embedding
      - https://arxiv.org/abs/1405.4053
    - 2015 – skip-thought vectors
      - Predict previous / next sentence with seq2seq model)
      - https://arxiv.org/abs/1506.06726
    - 2015 – auto-encoders (semi or unsupervised)
      - https://arxiv.org/abs/1511.01432 ,
      - https://arxiv.org/pdf/1511.06349.pdf,
      - https://arxiv.org/abs/1602.03483

23

Text classification – indexing

Learning word representations (vectors)
- How? -> NN-based
  - Word-level (2003 - …)
  - Sentence (document) level (2014 - …)
  - Contextual word-vectors (Word vectors compress all contexts into a *single vector)* *(2016 - …)*
    - 2016 context2vec https://www.aclweb.org/anthology/K16-1006.pdf
    - 2017 tagLM https://arxiv.org/abs/1705.00108
    - 2017 CoVe https://papers.nips.cc/paper/2017/hash/20c86a628232a67e7bd46f76fba7ce12-Abstract.html
    - 2018 ELMo https://www.aclweb.org/anthology/N18-1202/
    - 2018 ULMFiT https://arxiv.org/abs/1801.06146
    - 2019 BERT https://arxiv.org/abs/1810.04805



  - cross-lingual pre-training https://arxiv.org/abs/1706.04902
    - Code example https://github.com/huggingface/naacl_transfer_learning_tutorial
    - See also https://machinelearningmastery.com/use-word-embedding-layers-deep-learning-keras/

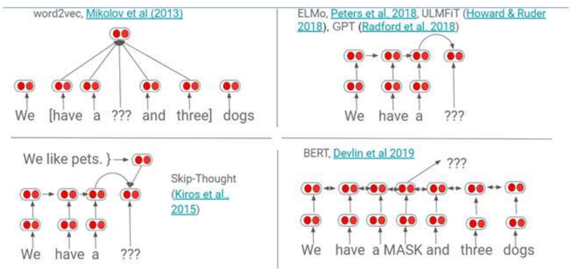
24

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification – indexing

Learning word representations (vectors)

- How? -> NN-based



25

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification

- Process
  - Data training analysis
    - Document indexing
      - Construct a document representation
        - Attributes / features
        - Weights of these features
      - Reduce the dimension of these representations
        - Feature selection
        - Feature extraction
    - Learning a classification model
  - Classification of test data
    - Text indexing
    - Apply the learnt model -> categories

26

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Text classification

- Reducerea dimensiunii
  - Are drept scop
    - Creşterea eficacităţii
    - Reducerea timpului de învăţare a modelului de clasificare
    - Evitarea învăţării pe derost a modelului de clasificare
  - Poate consta în
    - Selecţia atributelor (*feature selection*)
      - o submulţime a atributelor iniţiale (originale)
    - Extragerea atributelor
      - o mulţime de noi atribute determinate pe baza celor originale → proiecţia unui vector *R*-dimensional într-unul *r*-dimensional ($r < R$)
      - noile atribute (mai puţine) reprezintă o transformare a atributelor originale

27

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor

- Dându-se o mulţime de atribute $X_k=(x_{k1}, x_{K2},\dots,x_{km})$ pentru un document $d_k \epsilon D$, să se găsească o submulţime $X_K^p=(x_{K,i1}, x_{K,i2},\dots,x_{K,ip})$, cu $p < m$ care să optimizeze o funcţie obiectiv $J(X_K^m)$
  - Fc. obiectiv → eroarea de clasificare
- Selecţia implică
  - O strategie de căutare pentru selecţia submulţimilor candidat
    - căutare exhaustivă → toate submulţimile posibile →nefezabil
    - căutare strategică
      - prin ordonarea atributelor
        - pe baza unei metrici
        - şi alegerea celor care depăşesc un anumit prag
      - prin selectarea unei anumite submulţimi de atribute
        - se alege o submulţime optimală
  - O funcţie obiectiv pentru evaluarea acestor submulţimi candidat
    - măsură a calităţii unei submulţimi de atribute
    - ajută selecţia unei noi submulţimi candidat

28

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor

- Metode
  - Nesupervizate
    - Clusterizare
    - Factorizarea matricilor
  - Supervizate
    - Ordonarea atributelor
    - Selectia unei submultimi de atribute

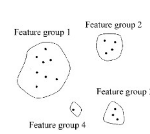
29

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Metode Nesupervizate → Clusterizare

- Se grupează atributele în clusteri
  - K-means
  - Hierarchical clustering



- Se înlocuiesc (multe) atribute similare din acelaşi cluster cu centrul clusterului

30

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Metode Nesupervizate → factorizarea matricilor

- Analiza componentelor principale
- Descompunerea in valori singulare
- Factorizarea matricilor non-negative
- Isomap-uri
- Self-organized maps (SOMs)

31

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin ordonarea atributelor

- Pp. că avem n date $(\mathbf{x}_k, y_k)$, $k=1,2,\ldots,n$
  - $\mathbf{x}_k \in \mathbf{R}^m \Rightarrow \mathbf{x}_k = (x_{k1}, x_{k2}, \ldots, x_{km})$
  - $y_k \in \mathbf{R}$
- Se calculează o funcţie scor pentru fiecare pereche $S(i)=(x_{ki}, y_k)$
  - cu cât scorul este mai mare, cu atât variabila este mai importantă
- şi se ordonează atributele în funcţie de acest scor
- Notaţie
  $X_i \in \mathbf{R}^n \Rightarrow X_i=(x_{1i}, x_{2i}, \ldots, x_{ni})$
  $Y \in \mathbf{R}^n \Rightarrow Y=(y_1, y_2, \ldots, y_n)$

32

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin ordonarea atributelor

- Scoruri posibile
  - Coeficientul de corelaţie al lui Pearson
    - $R(i)=cov(X_i, Y)/(var(X_i)var(Y))^{1/2}$
    - $R(i)\approx\Sigma_{k=1,\ldots,n}(x_{k,i}- X_i^a)(y_k-Y^a)/(\Sigma_{k=1,\ldots,n}(x_{k,i}- X_i^a)^2\Sigma_{k=1,\ldots,n}(y_k-Y^a)^2)^{1/2}$
    - $R^2(i)$ → relaţie de dependenţă liniară între X_i şi Y
  - Eroarea de clasificare
    - Mai mulţi clasificatori cu o singură variabilă
      - (x_{ki}, y_k), $k=1,2,\ldots,n$
      - Se stabileşte eroarea de clasificare pt fiecare $i=1,2,\ldots,n$
      - Se ordonează variabilele în funcţie de eroare
      - Cu cât eroarea este mai mică cu atât variabila este mai importantă
  - Informaţia teoretică
    - Informaţia mutuală între densitatea variabilei X_i şi densitatatea variabilei Y
    - $I(i)=\int_x\int_y p(x_i,y)\log(p(x_i,y)/(p(x_i)p(y)))dxdy$
    - $p(x)$ – probabilitatea densităţii lui x → greu de estimat

33

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin ordonarea atributelor

- Critici
  - poate determina submulţimi de atribute redundante
  - nu ţine cont de corelarea atributelor
  - un atribut nefolositor în izolaţie poate fi util în combinaţie cu alte atribute

34

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin alegerea unei submulţimi de atribute

- Căutarea
  - Căutare exhaustivă – toate submulţimile posibile → nefezabilă
  - Căutare strategică – alegerea doar a unor submulţimi
- Funcţia obiectiv – tipuri
  - *Wrapper*
  - *Filter*
  - *Embedded*

35

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin alegerea unei submulţimi de atribute

- Funcţia obiectiv – tipuri
  - *Wrapper*
    - Funcţia obiectiv este un clasificator care evaluează fiecare submulţime prin puterea ei predictivă
    - Alegerea atributelor este **dependentă** de performanţa clasificatorului (algoritmului de învăţare)
    - Algoritmul de învăţare = cutie neagră pentru evaluarea submulţimii de atribute în funcţie de puterea de învăţare (clasificare) a acesteia
  - *Filter*
    - Funcţia obiectiv evaluează fiecare submulţime doar pe baza conţinutului ei
    - Alegerea atributelor este **independentă** de performanţa clasificatorului
    - Selecţia atributelor este un pas anterior învăţării
  - *Embedded*
    - Alegerea atributelor are loc **în timpul** învăţării

36

Feat/output	Numeric	Categorical
Numerical	Pearson (linear) Spearman (non-lin)	LDA Rank-based meth
Categorical	ANOVA (linear) Kendall (non-lin)	Chi-Square Mutual Information

37

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin alegerea unei submulţimi de atribute → *Wrapper*

- Ideea de bază
  - *Wrapper* → a înveli, a împacheta
  - Funcţia obiectiv este un clasificator care evaluează fiecare submulţime prin puterea ei predictivă
  - Alegerea atributelor este **dependentă** de performanţa clasificatorului (algoritmului de învăţare)
  - Algoritmul de învăţare = cutie neagră pentru evaluarea submulţimii de atribute în funcţie de puterea de învăţare (clasificare) a acesteia
- Algoritm
  - Se alege o metodă de clasificare (învăţare)
  - Se caută configuraţia optimă (submuţime de atribute şi parametri ai clasificatorului)
    - Se alege o submulţime de atribute
    - Se repetă
      - Învăţarea şi optimizarea clasificatorului
      - cuantificarea performanţei clasificatorului
      - alegerea unei noi submulţimi de atribute
    - până când se obţine cea mai bună performanţă în învăţare

38

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin alegerea unei submulţimi de atribute → *Wrapper*

- Cum se alege o submulţime?
  - *best-first*
  - *branch-and-bound*
  - *simulated annealing*
  - algoritmi genetici
  - *greedy*
    - *Forward selection*
      - Variabilele sunt încorporate progresiv în submuţimi tot mai mari
    - *Backward selection*
      - Variabilele sunt eliminate progresiv din submulţime
- Cum se stabileşte performanţa algoritmului de învăţare?
  - Validare
  - Validare-încrucişată
- Care algoritm de învăţare să se folosească?
  - Arbori de decizie
  - Reţele neuronale
  - Maşini cu suport vectorial
  - Algoritmi evolutivi, etc

39

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin alegerea unei submulţimi de atribute → Filter

- Ideea de bază
  - Funcţia obiectiv evaluează fiecare submulţime doar pe baza conţinutului ei
  - Alegerea atributelor este **independentă** de performanţa clasificatorului
  - Selecţia atributelor este un pas anterior învăţarii
- Evaluare
  - Distanţa sau măsura separabilităţii claselor
    - Ex. distanţa (Euclideană, Hamming, etc) între clase
  - Corelaţia şi măsuri de informaţie teoretică
    - Submulţimile bune conţin atribute
      - puternic corelate cu ieşirea
      - ne-corelate între ele
    - Măsuri liniare
      - Coeficientul de corelaţie
    - Măsuri neliniare
      - Informaţia mutuală

40

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Selecţia atributelor → Prin alegerea unei submulţimi de atribute

- http://jmlr.csail.mit.edu/papers/volume3/guyon03a/guyon03a.pdf
- http://jmlr.csail.mit.edu/proceedings/papers/v4/guerif08a/guerif08a.pdf
- http://courses.cs.tamu.edu/rgutier/cs790_w02/l5.pdf

41

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Analiza documentelor de antrenament
  - Indexarea documentelor
    - Construirea unei reprezentări a documentelor → transformarea documentelor într-o formă interpretabilă de către clasificator
      - Obţinerea unor concepte/termeni reprezentative(i) → atribute
      - Calcularea unor ponderi pt aceste atribute
    - **Reducerea dimensiunii (a numărului de concepte/atribute/termeni reprezentative(i) pentru document)**
      - Selecţia atributelor
      - **Extragerea atributelor**
  - Învăţarea unui model de clasificare
- Clasificarea noilor documente(de test)

42

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

Reducerea dimensiunii → Extragerea atributelor

- Definire
  - Determinarea unei noi mulţimi de atribute determinate pe baza celor originale → proiecţia unui vector R-dimensional într-unul r-dimensional ($r < R$)
  - Noile atribute (mai puţine) reprezintă o transformare a atributelor originale
- Dându-se o mulţime de atribute $X_k=(x_{k1}, x_{k2},\ldots,x_{km})$, să se găsească o transformare $z_k=g(x_k):R^m \rightarrow R^p$ cu $p < m$ astfel încât transformarea z_k să păstreze (cea mai parte din) informaţia atributelor iniţiale
  - Transformarea optimă – cea care nu determină creşterea probabilităţii de eroare
  - Transformarea poate fi
    - Liniară $y = Wx$, $W \in M_{m,p}$
    - Ne-liniară – greu de determinat
  - Transformarea este ghidată de o funcţie obiectiv care trebuie optimizată (min/max)
- Metode de extragere a atributelor în funcţie de criteriul măsurat de funcţia obiectiv:
  - Reprezentare a semnalului → transformarea are drept scop reprezentarea datelor cu o acurateţe cât mai bună într-un spaţiu mai redus
    - Analiza componentelor principale (PCA)
  - Clasificare → transformarea are drept scop evidenţierea discriminării între clase într-un spaţiu mai mic
    - Analiza discriminantului linear (LDA)
- Metode – in functie de relatia dintre features
  - Relatii liniare
    - PCA, LDA
  - Relatii neliniare
    - Locally linear embedding -> manifold learning
    - t-SNE
    - Auto-encoders

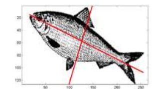
43

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Metode de reducere a dimensiunii → Extragerea atributelor → Analiza componentelor principale
  - Scop
    - Transformarea unui set de variabile posibil corelate într-un set de variabile necorelate între ele (componente principale)
    - Prima componentă principală are cea mai mare varianţă → cuantifică cea mai mare variabilitate posibilă a datelor
    - ACP determină axele care explică cel mai bine dispersia datelor (norul de puncte)
    - Descrierea datelor într-un spaţiu dimensional mai mic
  - Alte denumiri
    - Transformarea Karhunen-Loève

(teoria comunicaţiilor)



44

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

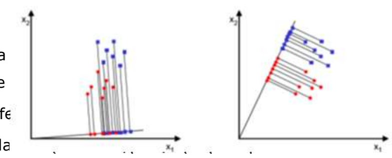
- Metode de reducere a dimensiunii → Extragerea atributelor → Analiza componentelor principale
  - Tipologie
    - ACP liniară – date separabile liniar
    - ACP bazată pe kernele – date neseparabile liniar
  - Algoritm
    - Pp că avem un set de date x_i, $i=1,2,..,n$ cu m atribute ($x_i \in R^m \Rightarrow x_i=(x_{i1}, x_{i2},\ldots,x_{im})$)
    - Scăderea mediei din fiecare dată (pe fiecare dimensiune) → centrarea datelor
      - $x'_{ij}=x_{ij} - x_j^a$, unde $x_j^a= (x_{1j}+x_{2j}+\ldots+x_{nj})/n$
    - Calcularea matricii de covariaţie C
      - $C = (c_{ij})$, $i, j =1,2,\ldots,m$, $c_{ij} = cov(x_i, x_j)$, unde $x_i=(x_{1i},x_{2i},\ldots,x_{ni})$
      - $cov(X,Y)=\Sigma_{i=1,2,\ldots,n}(X_i-X_a)(Y_i-Y_a)/(n-1)$
    - Determinarea vectorilor proprii $\boldsymbol{v}_p$ şi a valorilor proprii v_p (*eigenvector, eigenvalue*) corespunzătoare matricii de covariaţie $A\ \boldsymbol{v}_p= v_p\ \boldsymbol{v}_p$
    - Alegerea componentelor şi formarea vectorului de caracteristici (atribute)
      - Se ordonează vectorii proprii descrescător după valorile proprii → atributele în ordinea importanţei
      - Formarea vectorului de caracteristici cu acei vectori proprii care se doresc a fi reţinuţi
    - Derivarea noilor date
      - Se înmulţeşte vectorul de caracteristici cu vectorul datelor centrate

45

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Metode de reducere a dimensiunii → Extragerea atributelor → Analiza discriminantului liniar
  - Scop
    - Determinarea
      separe datele
    - Modelarea dife
    - Proiectarea da
      observa o mai bună separabilitate a datelor → care este cea mai bună proiecţie?
      - $y = w^T x$

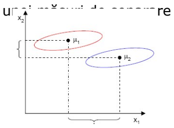


46

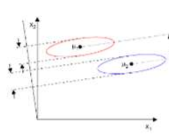
UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Metode de reducere a dimensiunii → Extragerea atributelor → Analiza discriminantului liniar
  - Găsirea celei mai bune proiecţii necesită definirea u... re între proiecţiile datelor
    - Distanţa între proiecţiile mediilor corespunzătoare datelor din fiecare clasă
      - Nu este foarte bine pentru că nu se ţine cont de dispersia datelor în interiorul claselor



    - Fisher → maximizarea raportului dintre diferenţa mediilor şi împrăştierea în interiorul claselor
      - → o proiecţie astfel încât:
        - exemplele din aceeaşi clasă sunt proiectate foarte aproape unele de altele
        - proiecţiile mediilor fiecărei clase sunt cât mai depărtate unele de altele



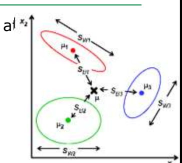
47

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

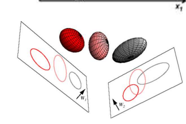
Clasificarea automată a textelor – Învăţare – proces

- Metode de reducere a dimensiunii → Extragerea a

Analiza discriminantului liniar



  - Algoritm
    - Pp că:
      - există k clase,
      - μ_i – media instanţelor din clasa i, $i=1,2,\ldots,k$
      - n – nr total de instanţe
      - n_i – nr de instanţe din clasa i, $i=1,2,\ldots,k$
    - *Se caută k-1 vectori de proiecţie*
    - Se calculează
      - Împrăştierea intra-clasă (scatter within class) S_w
        - $S_w=\Sigma_{i=1,2,\ldots,k}\Sigma_{x \in clasa i}(x-\mu_i)(x-\mu_i)^T$
      - Împrăştierea între clase (scatter between classes) S_b
        - $S_b=\Sigma_{i=1,2,\ldots,k}n_i(\mu_i-\mu)(\mu_i-\mu)^T$, unde $\mu=1/n\Sigma_{x \in clasa i}n_i\mu_i$
    - Se maximizează
      - Raportul dintre
        - Pătratul diferenţei mediilor (claselor) şi
        - Împrăştierea intra-clasă
    - Soluţie
      - $w=S^{-1}_w(\mu_1-\mu_2)$



- http://research.cs.tamu.edu/prism/lectures/pr/pr_l10.pdf
- http://www.dtreg.com/lda.htm
- http://www.music.mcgill.ca/~ich/classes/mumt611_05/classifiers/lda_theory.pdf

48

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Analiza documentelor de antrenament
  - Indexarea documentelor
    - Construirea unei reprezentări a documentelor → transformarea documentelor într-o formă interpretabilă de către clasificator
      - Obţinerea unor concepte/termeni reprezentative(i) → atribute
      - Calcularea unor ponderi pt aceste atribute
    - Reducerea dimensiunii (a numărului de concepte/atribute/termeni reprezentative(i) pentru document)
      - Selecţia atributelor
      - Extragerea atributelor
  - **Învăţarea unui model de clasificare**
- Clasificarea noilor documente(de test)
  - Indexarea documentelor
  - Utilizarea modelului de clasificare pentru stabilirea categoriilor fiecărui document de test

49

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Învăţarea unui model de clasificare
  - Alegerea unui algoritm de învăţare
    - Arbori de decizie
    - Reţele neuronale artificiale
    - Maşini cu suport vectorial
    - Algoritmi evolutivi
    - Reţele Bayesiene
  - Fixarea/optimizarea parametrilor algoritmului
    - Cum se aleg parametrii?
  - Construirea modelului de clasificare şi salvarea lui

50

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Analiza documentelor de antrenament
  - Indexarea documentelor
    - Construirea unei reprezentări a documentelor → transformarea documentelor într-o formă interpretabilă de către clasificator
      - Obţinerea unor concepte/termeni reprezentative(i) → atribute
      - Calcularea unor ponderi pt aceste atribute
    - Reducerea dimensiunii (a numărului de concepte/atribute/termeni reprezentative(i) pentru document)
      - Selecţia atributelor
      - Extragerea atributelor
  - Învăţarea unui model de clasificare
- **Clasificarea noilor documente(de test)**
  - Indexarea documentelor
  - Utilizarea modelului de clasificare pentru stabilirea categoriilor fiecărui document de test

51

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Clasificarea automată a textelor – Învăţare – proces

- Metode de reducere a dimensiunii
  - Extragerea atributelor
    - Analiza componentelor principale
    - Analiza componentelor independente
    - Scalare multidimensională
    - Hărţi topografice

52

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

- Text to numbers
  - https://machinelearningmastery.com/prepare-text-data-deep-learning-keras/
- Word embedding
  - https://www.tensorflow.org/tutorials/word2vec
  - https://nlp.stanford.edu/projects/glove/
  - https://machinelearningmastery.com/use-word-embedding-layers-deep-learning-keras/
  - https://machinelearningmastery.com/develop-word-embeddings-python-gensim/

53

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

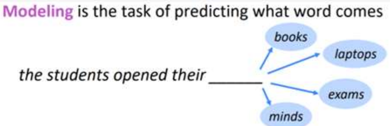
Natural Language Processing

- Text classification
- **Language modeling**
- Machine translation
- …

54

Language modelling

- **Language Modeling** is the task of predicting what word comes next.



- More formally: given a sequence of words $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(t)}$, compute the probability distribution of the next word $\boldsymbol{x}^{(t+1)}$:

$$P(\boldsymbol{x}^{(t+1)} | \ \boldsymbol{x}^{(t)}, \ldots, \boldsymbol{x}^{(1)})$$

where $\boldsymbol{x}^{(t+1)}$ can be any word in the vocabulary $V = \{\boldsymbol{w}_1, ..., \boldsymbol{w}_{|V|}\}$

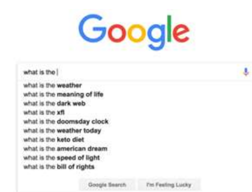
- A system that does this is called a **Language Model**.

55

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- Practical tasks
  - Sentence or document classification (e.g. sentiment)
  - Sentence pair classification (e.g. NLI, paraphrase)
  - Word level (e.g. sequence labeling, extractive Q&A)
  - Structured prediction (e.g. parsing)
  - Generation (e.g. dialogue, summarization, machine translation)
    - See https://medium.com/@samim/obama-rnn-machine-generated-political-speeches-c8abd18a2ea0



56

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- Formal task
  - Assign a probability to a sentence
    - Machine Translation:
      - P(**high** winds tonite) > P(**large** winds tonite)
    - Spell Correction
      - The office is about fifteen **minuets** from my house
        - P(about fifteen **minutes** from) > P(about fifteen **minuets** from)
    - Speech Recognition
      - P(I saw a van) >> P(eyes awe of an)
    - + Summarization, question-answering, etc., etc.!!

57

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- Datasets
  - https://nlpprogress.com/english/language_modeling.html
- Evaluation
  - Extrinsic
    - Run two models on the same task and compare the accuracies
    - +: simple, relevant
    - -: time consuming
  - Intrinsic
    - Perplexity
      - measure how well a probability distribution predicts a sample
      - Remember: Cross-entropy (H) -> classification performance (max)
      - Perplexity = 2^H (lower = better)
        - The Shannon Game:
          - How well can we predict the next word?

            I always order pizza with cheese and ____

            The 33rd President of the US was ____

            I saw a ____

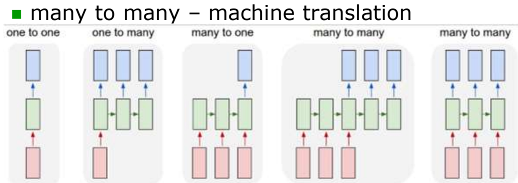
            mushrooms 0.1
            pepperoni 0.1
            anchovies 0.01
            ....
            fried rice 0.0001
            ....
            and 1e-100
          - Unigrams are terrible at this game. (Why?)
        - A better model of a text
          - is one which assigns a higher probability to the word that actually occurs

58

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - 1 to 1 – name entity recognition
  - 1 to many – image captioning
  - many to one – sentiment classification in sentences
  - many to many – machine translation

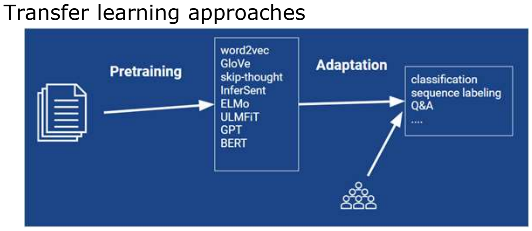


59

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - Traditional machine learning approaches
  - Transfer learning approaches



60

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - Count-based n-gram models
    - Approximate the history of observed words with just the previous n words
  - Neural n-gram models
    - Feedforward neural networks
      - Embed the n-gram history into a continuous space -> better capture the correlations between histories
    - Recurrent neural networks
      - Compress the entire history in a fixed length vector -> enable long range correlations

61

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How? -> Count-based n-gram models
  - Approximate the history of observed words with just the previous n words
  - Markov chains
    - Only previous history matters
    - Limited memory = previous k – 1 words
    - E.g. a 3-gram model:
      - a third order Markov model:
        - p(w1,w2, .., wn) = p(w1) p(w2 | w1) p(w3 | w1,w2) p(w4 | w2,w3)… p(wn | wn-2, wn-1)
      - Probability estimation:
        - p(w3 | w1, w2) = count(w1,w2,w3) / count (w1, w2)
  - + scalable, able to be trained on trillions of words
  - + fast constant time evaluation of probabilities at test time
  - + smoothing methods for matching the empirical distribution of language (heaps' law)
  - - large n-grams are sparse => hard to capture long correlations
  - - symbolic similarity does not involve semantic similarity (dog vs. cat)

62

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

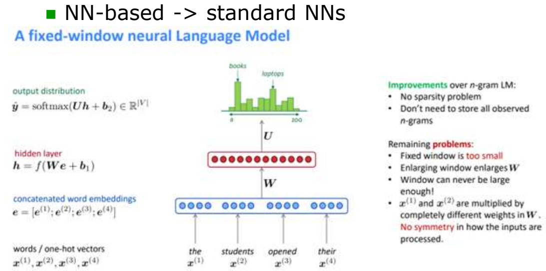
- How? -> Neural n-gram models -> Feedforward neural networks
  - Embed the n-gram history into a continuous space -> better capture the correlations between histories

63

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - NN-based -> standard NNs

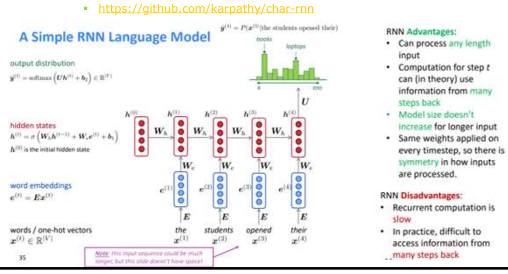


64

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - NN-based -> recurrent NNs
    - Standard NN versus Recurrent NN
      - https://www.deeplearningbook.org/contents/rnn.html
      - http://karpathy.github.io/2015/05/21/rnn-effectiveness/
    - Char-level recurrent NN
      - https://github.com/karpathy/char-rnn



65

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - NN-based -> recurrent NNs

RNN Disadvantages:
- Recurrent computation is slow
- In practice, difficult to access information from many steps back

Effect of vanishing gradient on RNN-LM

- **LM task:** *The writer of the books* ___ *is* / *are*
- **Correct answer**: *The writer of the books is planning a sequel*
- Syntactic **recency:** *The writer of the books is* (correct)
- Sequential **recency:** *The writer of the books are* (incorrect)

LSTM

66

Language modelling

- How?
  - NN-based -> special recurrent NNs
    - LSTM (2013-2015) –
      - https://www.bioinf.jku.at/publications/older/2604.pdf
      - http://colah.github.io/posts/2015-08-Understanding-LSTMs/

On step t, there is a hidden state $h^{(t)}$ *and* a cell state $c^{(t)}$

- Both are vectors length n
- The cell stores long-term information
- The LSTM can erase, write and read information from the cell

The selection of which information is erased/written/read is controlled by three corresponding gates

- The gates are also vectors length n
- On each timestep, each element of the gates can be open (1), closed (0), or somewhere in-between.
- The gates are dynamic: their value is computed based on the current context

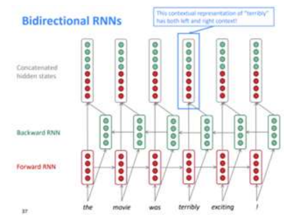
    - Gated Recurrent Units (2014) https://arxiv.org/pdf/1406.1078v3.pdf
      - Better speed and few parameters

67

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

Language modelling

- How?
  - NN-based -> fancy recurrent NNs
    - Bidirectional RNNs
      - Transformers (2017)
        - https://arxiv.org/abs/1706.03762
        - https://github.com/tensorflow/tensor2tensor
        - https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html
        - https://jalammar.github.io/illustrated-transformer/
      - BERT (Bidirectional Encoder Representations from Transformers)
        - https://arxiv.org/pdf/1810.04805.pdf
        - https://github.com/google-research/bert
        - https://www.youtube.com/watch?v=-9vVhYEXeyQ
    - Multi-layer RNNs



68

UNIVERSITATEA BABEŞ-BOLYAI
Facultatea de Matematică şi Informatică

- More details:
  - Cross-lingual transfer learning
    - https://ruder.io/talks/
    - https://docs.google.com/presentation/d/1fIhGikFPnb7G5kr58OvYC3GN4io7MznnM0aAgadvJfc/edit#slide=id.g5a24b37ba6_3_12
  - https://nlpprogress.com/

69