What is Relevant in a Text Document?

An Interpretable Machine Learning Approach

Leila Arras, Franziska Horn, Grégoire Montavon,
Klaus-Robert Müller, Wojciech Samek

Christoph Schaller

Word2Vec

Word2Vec

- Is an approach to learn vector representations of words
- Using the context words to create the initial vectors

Skipgram

- Is better to represent infrequent words
- Nearby context words have higher weight
- Trained by each context against the word

CBOW

- Predicts a word given a window of context words
- Order of context words has no weight
- Trained by each word against its context

Layer-Wise Relevance Propagation

Identifying Relevant Words

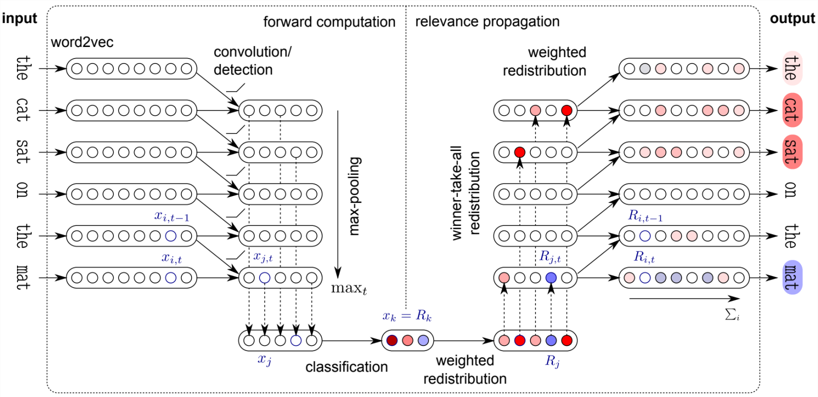

Figure 1: Diagram of a CNN-based interpretable machine learning system

Identifying Relevant Words

Needs a vector-based word representation and a neural network

Step One
Compute input representation of text document

Step Two
Forward-propagate input representation

Step Three
Backward-propagate using the layer-wise relevance propagation

Step Four
Pool the relevance score onto the input neurons

Step One

Computing the input representation of a text document

Words	Vectors
the	[0.035, -0.631, ...
cat	[0.751, -0.047, ...
sat	[0.491, 0.002, ...
on	[-0.181, -0.086, ...
the	[0.035, -0.631, ...

Table 1: CBOW vector example

Step Two

Forward-propagate the input representation until the output is reached

- We begin with our D×L matrix-representation of the document
  - D is the embedding dimension
  - L is the document size
- The convolutional layer produces a new representation of F features maps of length L - H + 1
- ReLU is applied element wise
- Features maps are pooled by computing the maximum over the text sequence of the document
- The maxpooled features are fed into a logistic classifier

Step Two

ML Model	Test Accuracy (%)
CNN1 (H=1, F=600)	79.79
CNN2 (H=2, F=800)	**80.19**
CNN3 (H=3, F=600)	79.75

Table 2: Performance of different CNN models

Step Three

Backward-propagate using the layer-wise relevance propagation

- Delivers one scalar relevance value per input variable, input data point and possible target class
- Redistributes the score of a predicted class back to the input space
- The Neuron that had the maximum value in the pool is granted all the relevance

Step Four

Pool the relevance score onto the convolutional layer

- $R_{(i,t-\tau)\leftarrow(j,t)} = \frac{z_{i,j,\tau}}{\sum_{i,\tau} z_{i,j,\tau}}$
- Similar to the Equation used for LRP
- More complex due to the convolutional structure of the layer

Pool the relevance score onto the input neurons

- $R_{i,t} = \sum_{j,\tau} R_{(i,t)\leftarrow(j,t+\tau)}$
- The Word that had the maximum value in the pool is granted all the relevance

Identifying Relevant Words

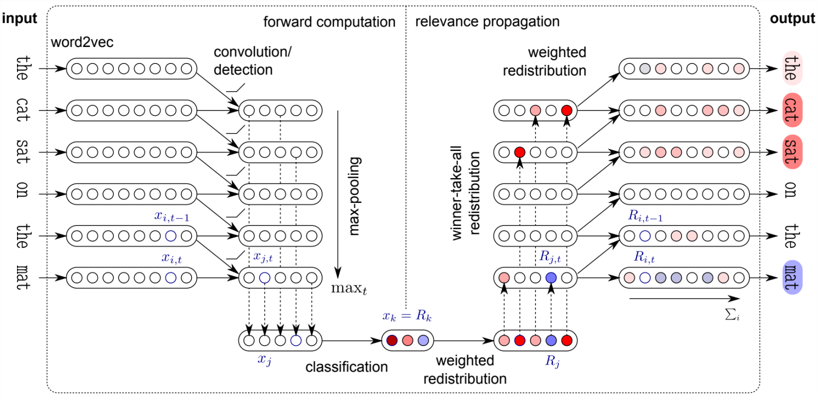

Figure 2: Diagram of a CNN-based interpretable machine learning system

Condensing Information

Obtaining relevance over all dimensions of word2vec

- $R_t = \sum_i R_{i,t}$
  pool relevances over all dimensions
- $\forall_i : d_i = \sum_t R_t \cdot x_{i,t}$
  condense semantic information
- $\forall_i : d_i = \sum_t R_{i,t} \cdot x_{i,t}$
  build document summary vector without pooling

BoW/SVM as Baseline

Bag of Words

- Documents are represented as vectors
- Each entry is TFIDF of a word in the training vocabulary

Support Vector Machine

- Hyperplanes are learned to separate classes
- Linear prediction scores for each class are obtained
- $s_c = w_c^\top x + b_c$
- w_c are class specific weights
- b_c is class specific bias

Performance Comparison

ML Model	Test Accuracy (%)
BoW/SVM (V=70631 words)	80.10
CNN1 (H=1, F=600)	79.79
CNN2 (H=2, F=800)	**80.19**
CNN3 (H=3, F=600)	79.75

Table 3: Performance of different ML Models

BoW/SVM as Baseline

LRP Decomposition

- $R_i = (w_c)_i \cdot x_i + b_c/D$
- D is the number of non-zero entries of x

Vector Document Representation

- d is built component-wise
- $\forall_i : d_i = R_i \cdot \tilde{x}_i$
- Replacing R_i with a TFIDF score allows comparability

Why is this Approach the Baseline?

- Relies on word frequencies
- All words in the embeddings are equidistant

Quality of Word References

Comparing Relevance Scores

How to compare relevance scores assigned by algorithms?

Intrinsic Validation

Counting Words	Deleting Words
Creating a list of the most relevant words for a category across all documents	Removing words and measuring the decrease of the classification score

The second approach grants an objective estimation to compare relevance decomposition methods

Measuring Model Explanatory Power

How to compare the explanatory power of ML models?

Extrinsic Validation

Problems

- Need for common evaluation basis
- Classifiers differ in their reaction to removed words

Approach

Comparing models by how 'semantic extractive' their word relevances are

Measuring Model Explanatory Power

How to compare the explanatory power of ML models?

Step One

Compute document summary vectors for all test set documents

Step Two

- Normalize document summary vectors to euclidean norm
- perform K-nearest neighbor classification

Step Three

- Repeat Step Two over ten random data splits
- Average KNN classification accuracies

The maximum KNN accuracy is used as explanatory power index

Results

Identification of Relevant Words

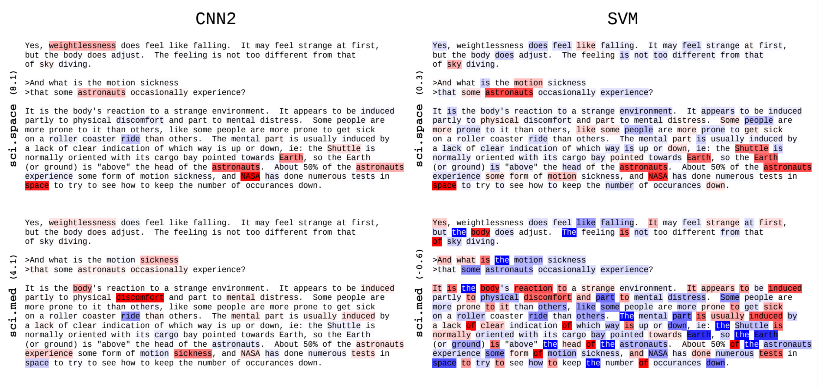

Figure 3: LRP heatmaps, positive is red, negative is blue

Identification of Relevant Words

sci.med	sci.space	comp.graphics
symptoms (7.3), treatments (6.6), medication (6.4), osteopathy (6.3), ulcers (6.2), sciatica (6.0), hypertension (6.0), herb (5.6), doctor (5.4), physician (5.1), Therapy (5.1), antibiotics (5.1), Asthma (5.0), renal (5.0), medicines (4.9), caffeine (4.9), infection (4.9), gastrointestinal (4.8), therapy (4.8), homeopathic (4.7), medicine (4.7), allergic (4.7), dosages (4.7), esophagitis (4.7), inflammation (4.6), arrhythmias (4.6), cancer (4.6), disease (4.6), migraine (4.6), patients (4.5).	spacecraft (11.0), orbit (10.8), NASA (8.6), Mars (7.8), moon (7.1), orbiting (7.1), Martian (6.8), orbital (6.8), shuttle (6.7), SMOS (6.6), telescope (6.5), Space (6.5), rocket (6.3), GRBs (6.0), Earth (6.0), astronaut (5.9), Moon (5.7), Shuttle (5.7), lander (5.6), Flyby (5.3), planets (5.2), Hubble (5.2), Soyuz (5.2), geosynchronous (5.2), Endeavour (5.1), space (5.0), planetary (4.9), Nasa (4.9), Astronomy (4.9), astronauts (4.9).	Graphics (6.9), raytracing (6.8), graphics (6.8), polygon (6.5), animation (6.3), Image (6.2), shaders (6.2), pixel (5.7), fractal (5.5), viewports (5.5), Autodesk (5.4), visualization (5.2), RGB (5.1), images (5.0), TIFF (5.0), Corel (4.9), Studio (4.9), algorithm (4.8), Bezier (4.8), polygons (4.7), GIF (4.7), Pixel (4.6), algorithms (4.5), modo (4.5), image (4.4), radiosity (4.4), AutoDesk (4.3), Studios (4.3), HPGL (4.2), JPEG (4.2).

Figure 4: The 30 most relevant words for CNN2

sci.med	sci.space	comp.graphics
cancer (1.4), photography (1.0), doctor (1.0), msg (0.9), disease (0.9), medical (0.8), sleep (0.8), radiologist (0.7), eye (0.7), treatment (0.7), prozac (0.7), vitamin (0.7), epilepsy (0.7), health (0.6), yeast (0.6), skin (0.6), pain (0.5), liver (0.5), physician (0.5), she (0.5), needles (0.5), dn (0.5), circumcision (0.5), syndrome (0.5), migraine (0.5), antibiotic (0.5), water (0.5), blood (0.5), fat (0.4), weight (0.4).	space (1.6), launch (1.4), ics.uci.edu (1.2), moon (1.1), orbit (1.0), mars (1.0), pat (1.0), nasa (0.9), dietz (0.9), shuttle (0.8), solar (0.7), command (0.7), henry (0.6), fred (0.6), gamma (0.6), sci.space (0.6), pluto (0.6), satellite (0.6), dc-x (0.6), nicho (0.6), astronomy (0.5), lunar (0.5), pom (0.5), hga (0.5), sky (0.5), spacecraft (0.5), gravity (0.5), scicom.alphacdc.com (0.5), nick (0.4), roland (0.4).	graphics (2.0), phigs (1.4), image (1.4), images (1.4), xv (1.3), tiff (1.2), polygons (1.1), comp.graphics (1.0), mpeg (1.0), format (1.0), siggraph (1.0), povray (0.9), quicktime (0.8), bockamp (0.8), surface (0.8), animation (0.8), iges (0.8), studio (0.8), jpeg (0.8), pov (0.7), dec (0.7), scodal (0.7), algorithm (0.7), genoa (0.7), sgi (0.7), palette (0.6), vga (0.6), impulse (0.6), c (0.6), rgb (0.6).

Figure 5: The 30 most relevant words for Bow/SVM

Document Summary Vectors

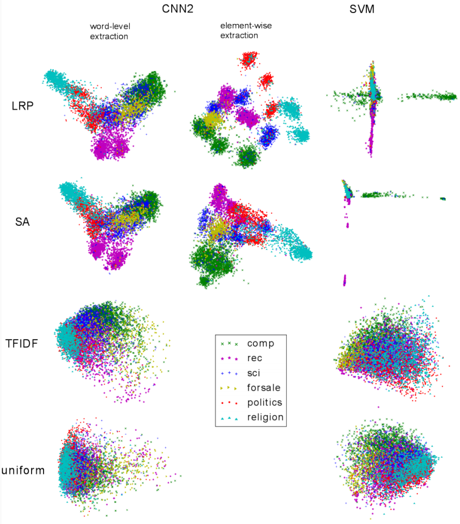

Figure 6: The 30 most relevant words for Bow/SVM

Evaluating LRP

How good is LRP in identifying relevant words?

- Delete Sequence of words from document
- Classify document again
- Report as function of accuracy and number of missing words

Evaluating LRP

Three different approaches

1. - Start with correctly classified documents
   - Delete words in decreasing order of their relevance
2. - Start with falsely classified documents
   - Delete words in increasing order of their relevance
3. - Start with falsely classified documents
   - Delete words in decreasing order of their score

Evaluating LRP

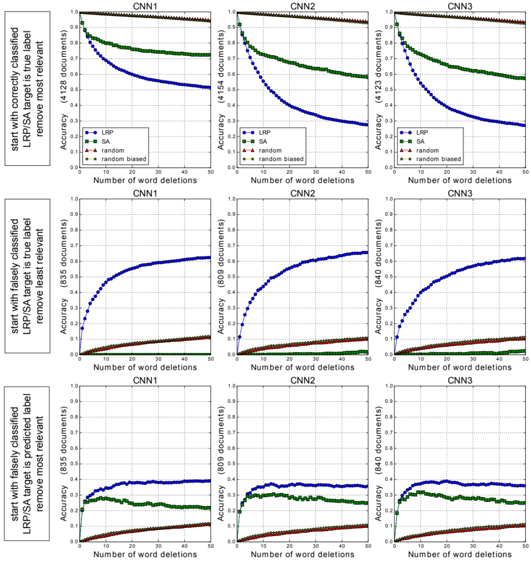

Figure 7: Word deletion experiments

Quantifying Explanatory Power

Semantic Extraction		Explanatory Power Index (EPI)	KNN parameter
word2vec/CNN1	LRP (ew)	0.8045 (± 0.0044)	K = 10
	SA (ew)	0.7924 (± 0.0052)	K = 9
	LRP	0.7792 (± 0.0047)	K = 8
	SA	0.7773 (± 0.0041)	K = 6
word2vec/CNN2	LRP (ew)	0.8076 (± 0.0041)	K = 10
	SA (ew)	0.7993 (± 0.0045)	K = 9
	LRP	0.7847 (± 0.0043)	K = 8
	SA	0.7767 (± 0.0053)	K = 8
word2vec/CNN3	LRP (ew)	0.8034 (± 0.0039)	K = 13
	SA (ew)	0.7931 (± 0.0048)	K = 10
	LRP	0.7793 (± 0.0037)	K = 7
	SA	0.7739 (± 0.0054)	K = 6
word2vec	TFIDF	0.6816 (± 0.0044)	K = 1
	uniform	0.6208 (± 0.0052)	K = 1
BoW/SVM	LRP	0.7978 (± 0.0048)	K = 14
	SA	0.7837 (± 0.0047)	K = 17
BoW	TFIDF	0.7592 (± 0.0039)	K = 1
	uniform	0.6669 (± 0.0061)	K = 1

Table 4: Results over 10 random data splits

Quantifying Explanatory Power

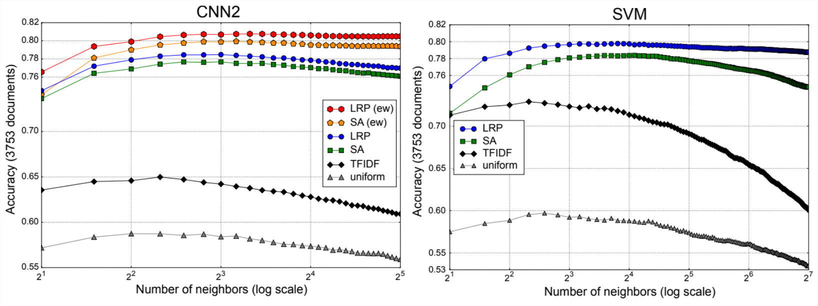

Figure 8: Word deletion experiments

Questions?