Bag-of-words Modelling for Speech Recognition

Bartosz Ziółko

Suresh Manandhar
Richard C. Wilson

www.dsp.agh.edu.pl

Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki
Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie

THE UNIVERSITY of York

Speech Recognition Scheme

Noise reduction

Segmentation

Parametrisation

Trained HMMs

Phoneme recogniser

Word level network

Dictionary

Sentence level network

Syntax model

Semantic coherence

LSA model

ASR of Polish

- Much fewer homophones
- More distinguishable sounds of vowels
- Very few irregularities in pronunciation
- Rustling phonemes
- Complicated grammar and morphology
- Inflective, not positional
- Possibly smaller dictionary

HMM Toolkit (HTK)

- Trained on 26 male adult speakers (9490 utterances)
- frequency 16 kHz
- 39 MFCCs
- 25 ms windows
- preemphasis filtering 0.97

Word – Topic Matrix

$$\mathbf{S} = [s_{ik}]$$

representing semantic relations, where rows $i = 1, ..., I$ represent topics and columns $k = 1, ..., K$ represent words. Each matrix value s_{ik} is the number of times word k occurs in topic i. A measure of similarity between two topics is

a dot product $$d_{ij} = \sum_{k=1}^{K} s_{ik} s_{jk}$$

Normalisation:

$$d'_{ij} = d_{ij} / \max_{i<j} \{d_{ij}\} \qquad 0 \leq d'_{ij} \leq 1$$

Training Algorithm (1)

Create an undirected, complete graph with topics as nodes and d'_{ij} as weights of edges. Let us define path weight

$$p_{ij} = \prod_{(a,b) \in P(i,j)} d'_{ab},$$

where $P(i, j)$ is the sequence of edges in the path from i to j. In the simplest case of a single edge i to j path weight is d'_{ij}. In case of a multiple edges path, it is a product of similarities of all edges on a path.

For each node, we need to find n nodes with highest measures of paths between the nodes and the given, analysed topic node. It will allow us to define a list N of semantically related topics which consists of the n nodes with their measures.

Training Algorithm (2)

The matrix S has to be recalculated to include impact of similar topics.Smoothed word-topic relations are expressed by matrix

$$\mathbf{S}' = [s'_{ik}].$$

For all topics in matrix we add all values of topics from the list of related topics, multiplied by a measure for a given pair of topics. The elements of S' are

$$s'_{ik} = s_{ik} + \alpha^{-1} \sum_{j \in N} p_{ij} s_{jk}.$$

Coefficient α is a smoothing factor which provides additional weight for influence of other topics on matrix S'.

Finding the Most Similar Topics

Find n single edge paths with the highest measures d'_{ij}.

Check if the two edges path $P(i, m)$ starting from the node i with the highest measure d'_{ij}, which was found in the step above and going through j to any other edge m, has a better measure p_{im} than the lowest of the n solutions found in the step above. If it does than replace the lowest one with m in the list of n similar topics.

Conduct the step above for all other single node paths from the list apart from the lowest, nth element.

If there are any non single edge paths $P(i, j)$ on the list on position different then nth, reapeat a process similar to step 2. Check if after adding any other edge, a measure of path p_{ij} is higher than a measure of the nth position. Than replace the previous path with a new longer one path with higher p_{ij}.

Semantic Model

Big John has a house. Big John has a black, aggressive cat. The black aggressive cat has a small mouse. The small mouse is a mammal.

All articles will be skipped as they have no semantic content and they do not exist in Polish which was our experimental language.

The first step of the algorithm is to count all other words in different topics, creating a matrix *S.*

Words-in-topics Matrix S

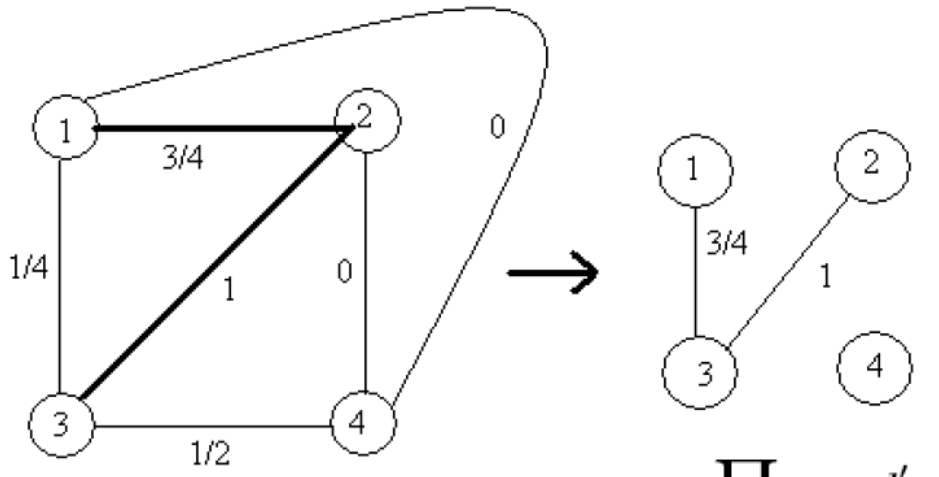
topic	big	John	has	house	black	aggr.	cat	small	mouse	is	mammal
1	1	1	1	1	0	0	0	0	0	0	0
2	1	1	1	0	1	1	1	0	0	0	0
3	0	0	1	0	1	1	1	1	1	0	0
4	0	0	0	0	0	0	0	1	1	1	1
3'	7/8	7/8	15/8	1/2	11/8	11/8	11/8	1	1	0	0

where rows $i = 1, \ldots, I$ represent topics and columns $k = 1, \ldots, K$ represent words. Each matrix value s_{ik} is the number of times word k occurs in topic i. A measure of similarity between two topics is

$$d_{ij} = \sum_{k=1}^{K} s_{ik} s_{jk}$$

$$d'_{ij} = d_{ij} / \max_{i<j} \{d_{ij}\}$$

Topic Similarities



$$p_{ij} = \prod_{(a,b)\in P(i,j)} d'_{ab}$$

topic	1	2	3	4
1	4	3	1	0
2	3	6	4	0
3	1	4	6	2
4	0	0	2	4

where *P*(*i*, *j*) is the sequence of edges in the path from *i* to *j*. In the simplest case of a single edge *i* to *j* path d'_{ij} ight is . In case of multiple edges path, it is a product of similarities of all edges on a path.

Modifying Matrix S

For each node, we need to find *n* nodes with highest measures of paths leading to them from the given node. That will allow us to define a list *N* of semantically related topics which consists of the *n* nodes with their measures.

$$\mathbf{S}' = [s'_{ik}] \qquad s'_{ik} = s_{ik} + \alpha^{-1} \sum_{j \in N} p_{ij} s_{jk}$$

3’	7/8	7/8	15/8	1/2	11/8	11/8	11/8	1	1	0	0

Recognition Using Semantic Model

Recognition can be conducted by finding the most coherent topic for a set of words *W* in a provided hypothesis. It is carried on by finding a maximum of a sum of elements of *S'* from columns representing the word [illegible]sis over rows

$$P_{sem} = \max_i \sum_{k \in W} s'_{ik}$$

The row *i*, for which the maximum is found is assumed to represent the topic of sentence being recognised.

$$P_{sem} \in \Re^+ \qquad p = p_{htk}^{w} p_{sem}$$

Preprocessing (SED)

1. Polish letters
2. Special signs , ” “ : ( ) ; + - n/ ’ # & ? !
3. Dots
    1. After an abbreviation, if it finishes with the finishing letter of the word
    2. Following digits if they mean order, like *th* in English

Corpora and Results

Training on Parliament Transcripts and Literature (around 100 million words)

Testing on 45 recorded sentences

n	α	w	ranking of the correct hypothesis	% improvement
LSA		30	12.36	-19
HTK			10.39	0
3	3	25	**8.95**	**14**

Thank you!

Mountains near Krakow, Poland on February