Classification of patent applications

Norbert Ryciak, Maciej Chrabąszcz, Maciej Bartoszuk

Sages

Problem

+	A	HUMAN NECESSITIES
+	B	PERFORMING OPERATIONS; TRANSPORTING
+	C	CHEMISTRY; METALLURGY
+	D	TEXTILES; PAPER
−	E	FIXED CONSTRUCTIONS
		BUILDING
−	E01	CONSTRUCTION OF ROADS, RAILWAYS, OR BRIDGES
−	E01B	**PERMANENT WAY; PERMANENT-WAY TOOLS; MACHINES FOR MAKING RAILWAYS OF ALL KINDS** (derailing or rerailing blocks on track, track brakes or retarders B61K; removal of foreign matter from the permanent way, vegetation control, applying liquids E01H)
		Structure of the permanent way for railways or tramways
	E01B 1/00	**Ballastway; Other means for supporting the sleepers or the track; Drainage of the ballastway** (draining by trenches, culverts, or conduits E01F 5/00) **[2006.01]**
	E01B 2/00	**General structure of permanent way** (railway networks B61B 1/00; foundations for pavings E01C 3/00; foundations in general E02D) **[2006.01]**
−	E01B 3/00	**Transverse or longitudinal sleepers** (for switches or crossings E01B 7/22)**; Other means resting directly on the ballastway for supporting rails [2006.01]**
−	E01B 3/02	• made from wood (drying or impregnating B27K) **[2006.01]**
−	E01B 3/04	• • Means for preventing cleaving **[2006.01]**

Problem characteristic

Data:

- Long documents (1 - few A4 pages)
- Specialized language
  - a lot of very rare words
- Structured documents (two parts: general and technical)
- Two types of documents: inventions and utility models
- Hidden time dimension

Problem characteristic

Task:

- Multilabel classification
  - huge number of classes (tens of thousands)
  - imbalanced dataset (including classes with only one example)
- Hierarchical classes
  - up to 11 levels in branch
- Classes with descriptions (potentially useful information)

State of the art NLP solutions

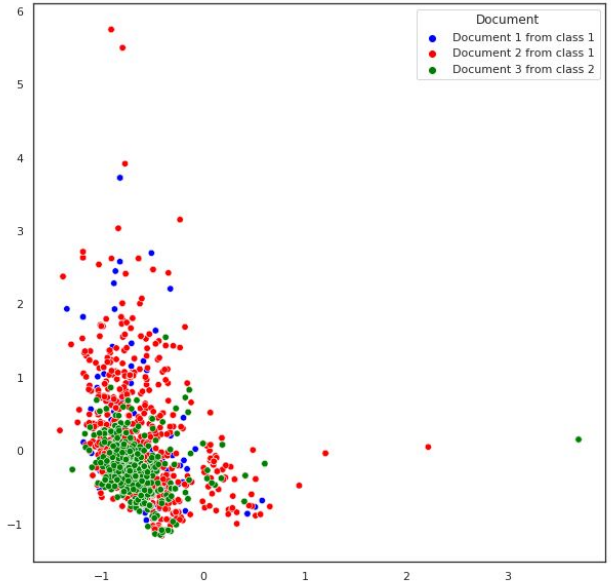
- Complex deep learning models,
- Based on embeddings,
- Treating document as sequences,
- Usually developed for short or medium lengths documents...

Tried and tested approaches

- Classical machine learning
- Doc2vec
- Classification based on document similarity
- Using class descriptions

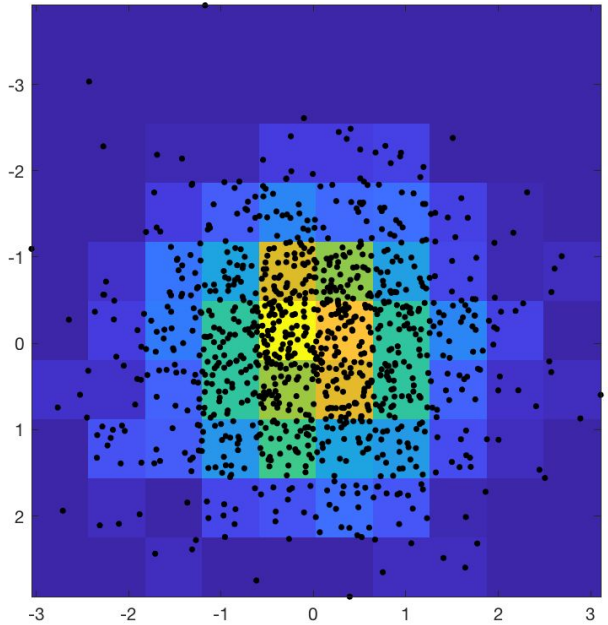
Final solution - “embeddings cloud”

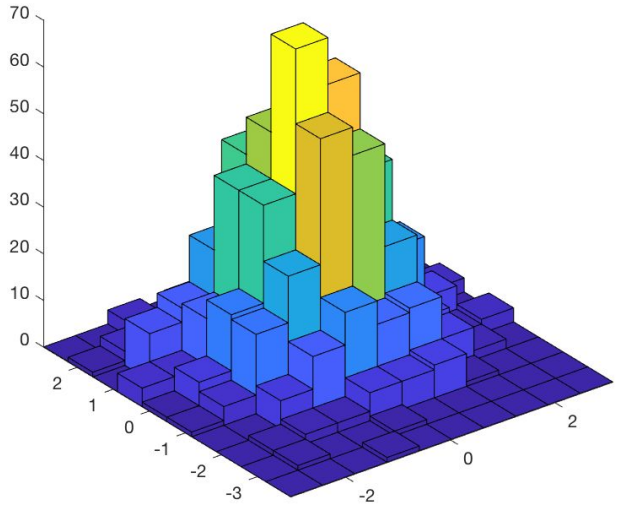
Motivation: distribution of word embeddings should be connected with category.



Embeddings cloud - how to represent distribution?

Representing two-dimensional distribution - histogram 2D

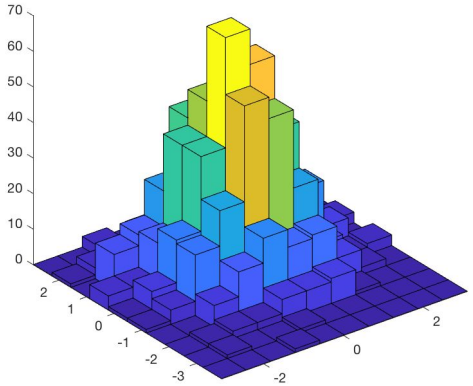


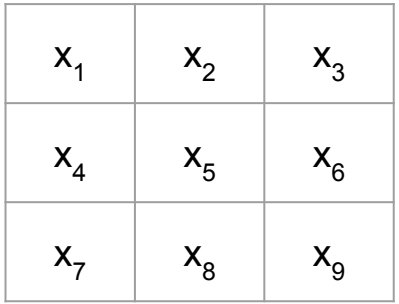


Source: https://www.mathworks.com/matlabcentral/fileexchange/66629-2-d-histogram-plot

Embeddings cloud

Representing two-dimensional distribution - histogram 2D





Feature vector

Source: https://www.mathworks.com/matlabcentral/fileexchange/66629-2-d-histogram-plot

But…

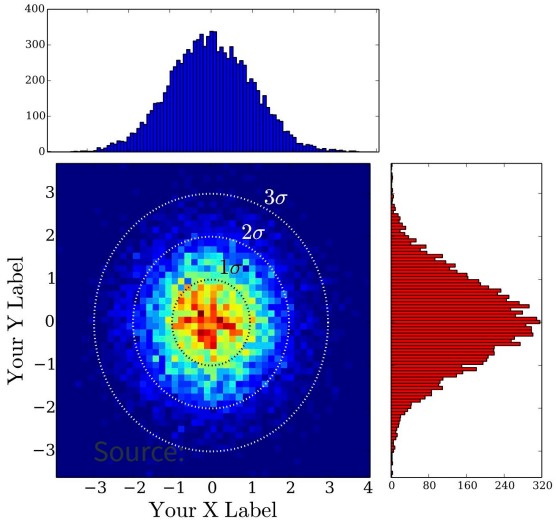
Best word embeddings are 300-dimensional.

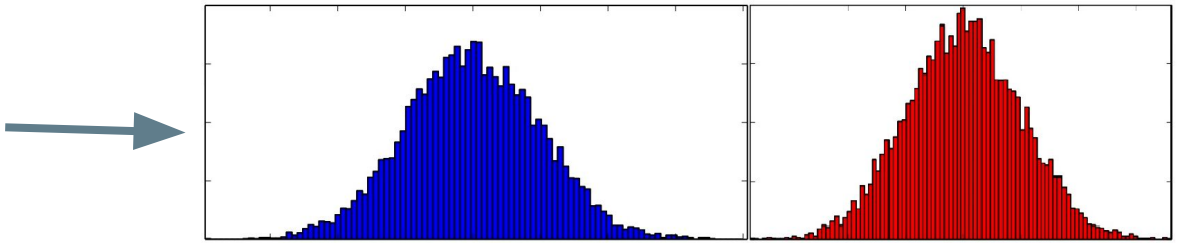
So if we use 10 bins per coordinate, dimensionality of 300D histogram will be…

10^{300}…

Solution

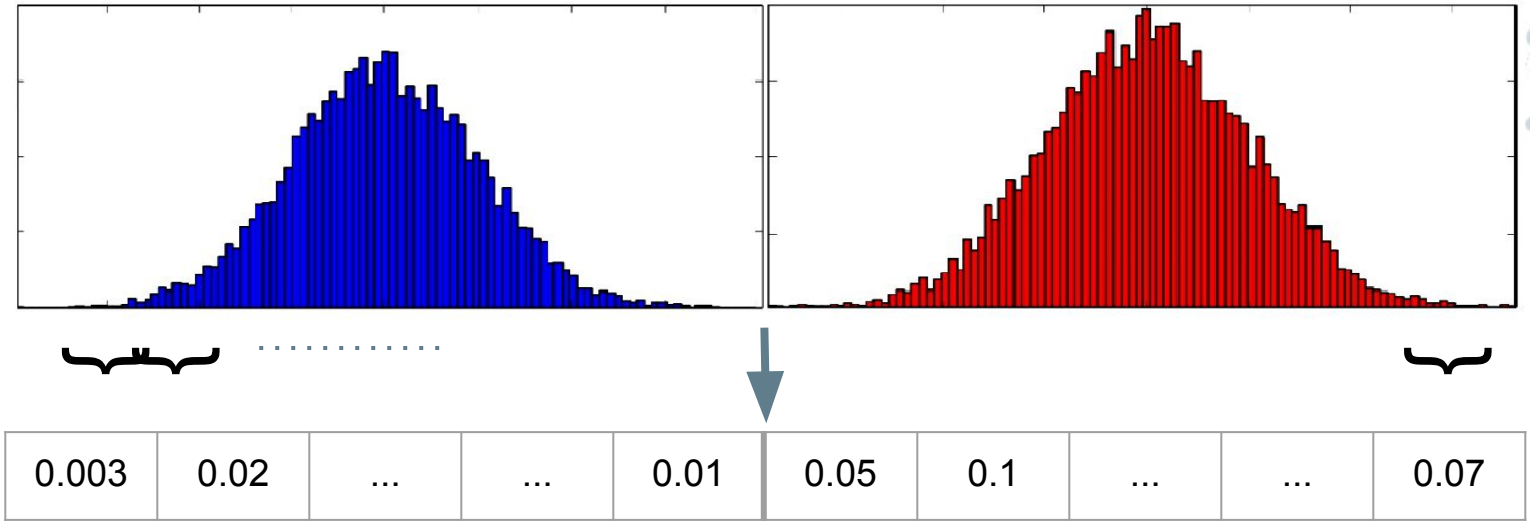
Instead of multi-dimensional distribution, we can describe distribution in simplified way - with the use of marginal distributions.





https://www.astrobetter.com/blog/2014/02/10/visualization-fun-with-python-2d-histogram-with-1d-histograms-on-axes/

Solution



Feature vector

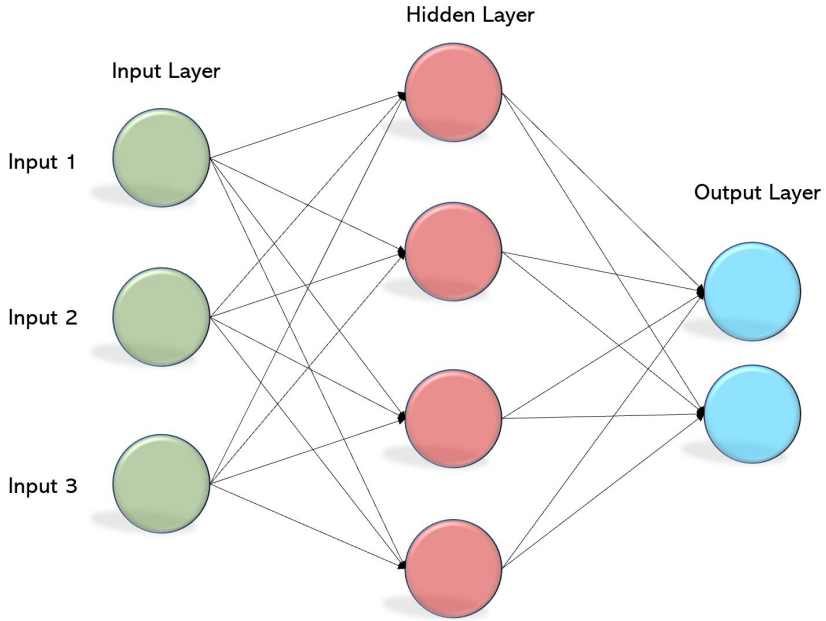
Then, using 10 bins per coordinate, dimensionality of final representation will be 300 * 10 = 3000!

Embedding cloud - features

- Reasonable representation for long documents,
  - if there is a lot of tokens in document distribution should refer to category,
- Reasonable representation for documents with specialized, rare words,
  - rare words related to specialized technical issues should have similar embeddings,
- Representation is constant-length vector.

Model

One-layer perceptron for multilabel classification.



Source:
https://becominghuman.ai/multi-layer-perceptron-mlp-models-on-real-world-banking-data-f6dd3d7e998f

Practical aspects

- We did not take into consideration structure of documents and type
- Number of bins was optimized (best: 30 per coordinate). So the final feature vector was 9000-dimensional
- Structure of network was optimized (finally 2096 neuron in hidden layer)
- Learning algorithm was optimized (Adagrad gave best results)
- We removed categories with less than 4 examples. It reduced number of labels from 90 000 to 40 000.
- We used data only from last 30 years (we tested: 10 - 50).

Competition vs real life

- Competition: very limited time (no time for developing complex deep learning models or greedy approaches based on documents similarity)

- Requirement for deployed solution: explainability!
  - System should explain its decision - especially point out key-words which led to the specific decision.

Results - 1st stage of competition

Evaluation rules:

- To each document propose 5 categories at 4th level.
- If at least 1 is valid, you get 1 point.

Our score: 56 points for 100 testing documents.

Results - 2nd stage of competition

Evaluation rules:

- To each document propose 5 categories (we consider all levels of categories).
- 3 best matched to true classes are taken into consideration.
- You get 1 point for guessing correct category at each level lower than 3rd level.

Our score: 39% of points for 200 testing documents (second place).

Thank you.