Russian Q&A Method Study: from Naïve Bayes to Convolutional Neural Networks

Kirill Nikolaev and Alexey Malafeev

National Research University Higher School of Economics,

Nizhny Novgorod, Russia

kinikolaev@edu.hse.ru; aumalafeev@hse.ru

AIST-2017 contribution

First results for the task:

- Linear SVM algorithm
  - 95% accuracy for English!
    - State-of-the-art results: Loni B. A survey of state-of-the-art methods on question classification. – 2011.
  - 68.7% accuracy for Russian…
    - **…despite 2158 questions dataset.**

Question typology example

Tag	Numeric Tag	Wording Examples
General	1	Что происходит в …? / What is happening in …?
Verification	2	Правда ли, что …? / Is it true that …?
Definition	3	Что означает/такое? / What is …? What does … mean?
Example	4	Приведи пример…? / Give an example of …?
Comparison	5	Чем похожи/отличаются…? / What are the similarities/differences between …?
Choice	6	X или Y? X or Y?
Concept completion	7	Кто? Что? Где? Когда? Куда? Откуда? Во сколько? Who? What? Where? When? What time?

Relevant studies and initial research

Text classification RCNN:

- Lai et.al. RCNN for Text Classification. – AAAI, 2015

  ❖ Human-designed features

  vs

  ❖ Unsupervised text classifier

- Only 9% acc. on our data:

  Decided to use CNN

Dataset modifications

60.22% - new baseline;

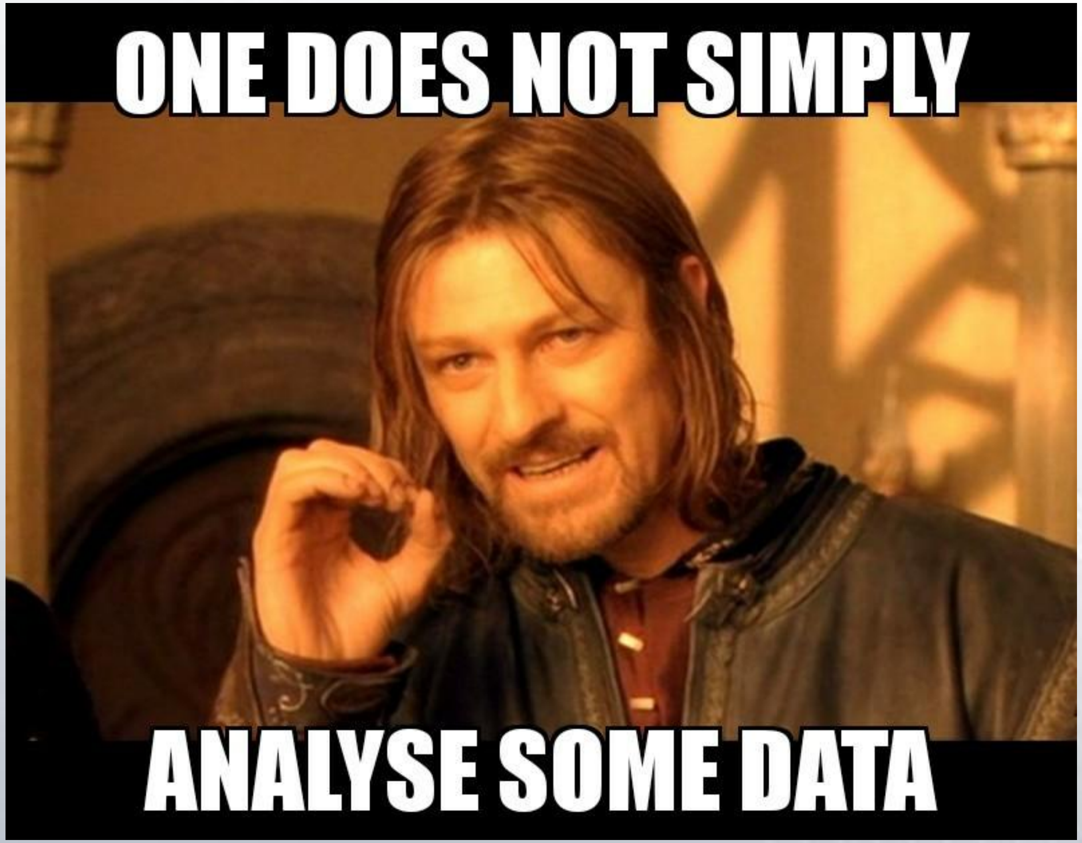
8.5% accuracy drop

10 & 11 merged;

2008 & 150 => 2086 & 181

4 & 5 boosted;

Data representation



Embedding approach – distributional semantics:

- Word2Vec
  - Pre-trained W2V model for Russian – Russian National Corpus (НКРЯ), 250 million words, 300-d vectors
  - Word: 300-d + 40 binary features

First 8 words pro sentence

- Average – 7;
- Resulting: 340x8

Architecture (https://github.com/Pythonimous/Q-A-System)

2-D Conv layer: 26 filters; kernel size: 20x3

Leaky ReLU: alpha = 0.1

MaxPooling2D

Dropout(0.2)

Flatten()

Dense(13, activation='softmax') – 72.38% test accuracy, 0.67 F-score

Additional experimenting

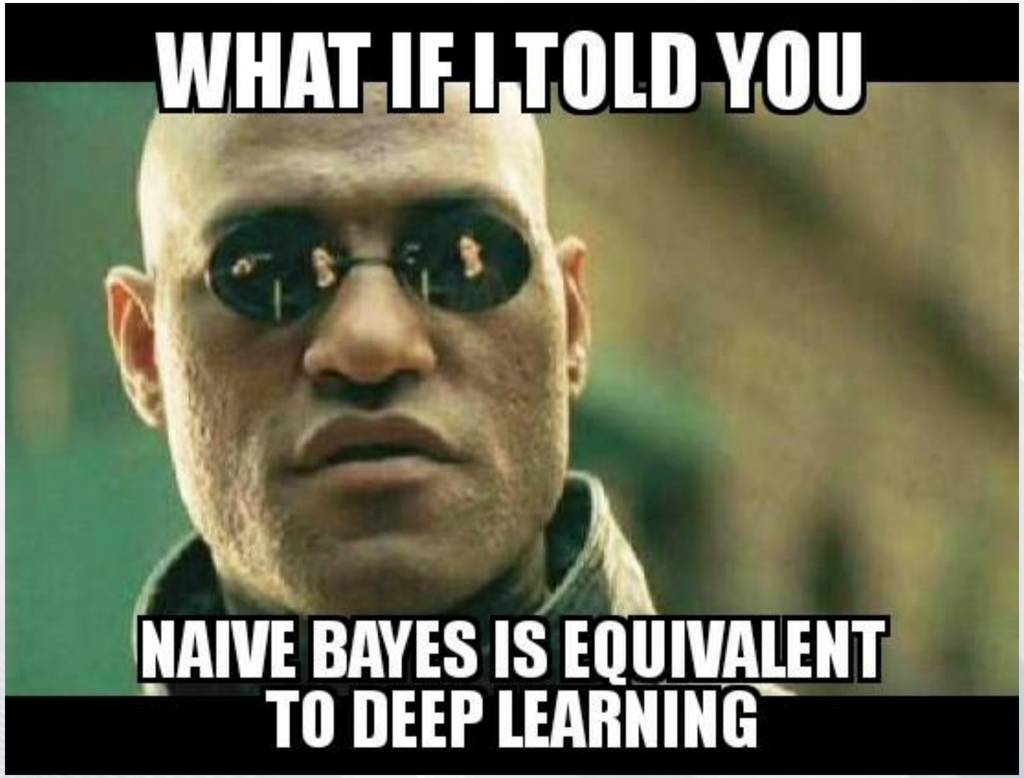
Small data: use Naïve Bayes?

- Only word features (Add-1):
  - 4912 and 3380 (lemmatized)
- Absolute frequency per question type

Word importance

- Counts replaced with PPMI (Add-2)

Top-1200 informative words: 70.72%
only slightly worse than CNN!



Words	All	2000	1500	1200	800	400	200
Accuracy	59.7%	62.4%	65.7%	70.7%	69.1%	68%	61.9%

Results and conclusions

Algorithm	Accuracy (micro)
2-D CNN	**72.38%**
Naïve Bayes (Top-1200)	70.72%
1-D CNN	68.61%
Trigram 1vsAll SVM **(Baseline)**	60.22%
Naïve Bayes (All words)	59.7%
Linear Regression	57%
RCNN, 3-D CNN	9%

- Quite possibly – the upper boundary for this dataset;
- Most problems: 1 (general), 10-11 (Action-Instrument);
- Possible improvements: dataset volume, more advanced (RCNN) algorithms and representations (3-D tensors)

Aspect-Based Sentiment Analysis of Russian Hotel Reviews

Valery Rybakov (valera210597@gmail.com)
Alexey Malafeev (aumalafeev@hse.ru)

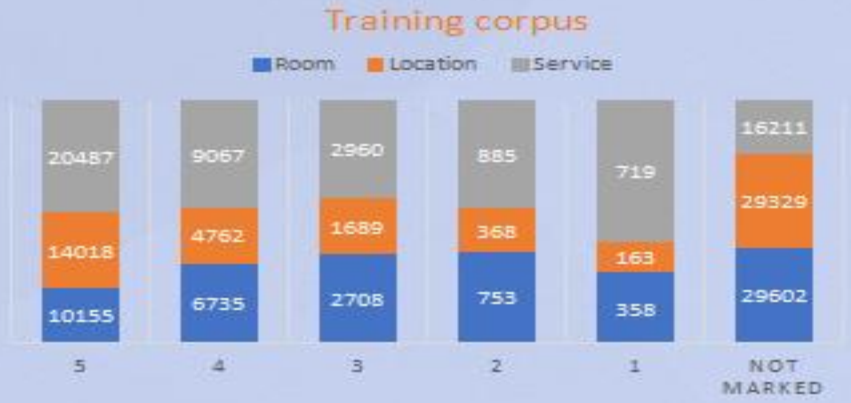
National Research University Higher School of Economics, Nizhny Novgorod, Russia

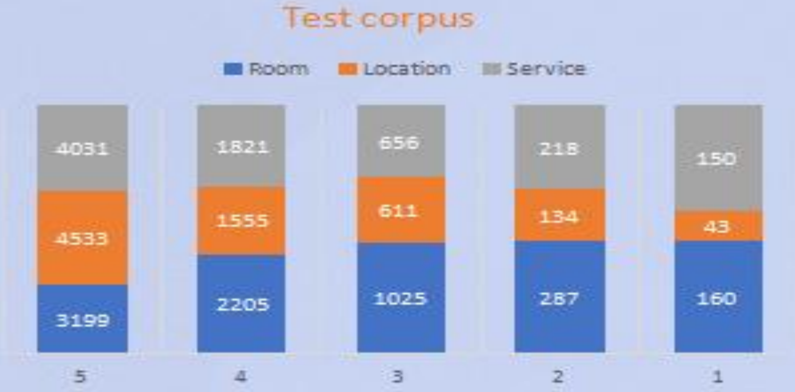
Aims and Methods

- The task of aspect-based sentiment analysis (ABSA) in the domain of Russian-language hotel reviews
- Based on the algorithm by ***Blinov and Kotelnikov*** **(2014, Dialogue)**
- The ***distributed representation of words*** was applied for constructing the aspect and sentiment lexicons.
- To build the vector space of words, a ***corpus*** comprising 57225 hotel reviews was constructed
- The lexicon construction approach was based on iteratively ***expanding*** a small set of ***initially specified terms***
- The sentiment of aspects in actual reviews was calculated given the ***aspect and sentiment terms*** found in the text and their ***weights***, i.e. cosine similarity to the initial terms

Corpus

- A new corpus of hotel reviews, collected from the website TripAdvisor.com, was assembled (57225 reviews), **see the link at the bottom**
- The following information was collected from the site:
  - the text of the review,
  - the overall rating of the hotel (on a 5-point scale),
  - an assessment of the hotel's characteristics: the price-quality ratio, location, room, cleanliness, service, quality of sleep





Normalization

* ***Review marks*** are deleted
* Texts are ***lemmatized*** (mystem) and ***segmented*** by sentences
* Each segment is ***tokenized***
* The ***punctuation marks*** are deleted
* ***Stop words*** are removed
* Collocation problem (pymorphy2 is used)

не/ очень+ next adjective/adverb/verb

a separate term

Terms extraction

For the vector space construction, ***word2vec*** was used

Initial term(s) for aspect *Room (as an example)*				
Номер	Ванная	Телевизор	Свет	Кровать

10 most similar terms for each initial term				
Номер	Ванная	Телевизор	Свет	Кровать
комната, прихожая, пространство...	раковина, душевая, санузел, ...	встроить, плоский, панель...	освещение, лампочка, спот...	прикроватный, диван, зеркало...

Combine lists, delete duplicates

For each term in a list repeat 2 and 3 steps until all words in the lexicon are processed

Number of terms for each aspect and sentiment				
Room	Location	Service	Positive	Negative
2550	1317	1740	342	1203

Aspect score calculation

* The review text is ***segmented*** by the following punctuation marks: {*? ! , . : ;*}
  * ***Weight*** of a term is the similarity between this term and the initial term(s)
    * For each segment, the aspect and sentiment terms and their weights are identified

\+ 0.2217 | Location 0.4484 | + 0.3089 | Location 0.1906

Отель *хорошо* **расположен,** (1) *рядом* много **магазинчиков,** (2)

Room 1.000 | - -0.2793

однако сам отель и **номера** довольно *старые* (3)

Sentiment value calculation for each aspect in the sentence

- (1) <u>Location</u>: 0.4484 * (0.2217 + 0.3089) +

  \+ (2) 0.1906 * (0.2217 + 0.3089 - 0.2793)
- (3) <u>Room</u>: 1 * (0.3089 - 0.2793 + 0.4642)
- (4) <u>Service</u>: 0.656 * (- 0.2793 + 0.4642)

The number of correct and incorrect decisions of the algorithm:

Room

Category		Actual class	
		Positive	Negative
Predicted class	Positive	3228	60
	Negative	2176	387

Location

Category		Actual class	
		Positive	Negative
Predicted class	Positive	4778	119
	Negative	1301	58

Service

Category		Actual class	
		Positive	Negative
Predicted class	Positive	4090	145
	Negative	1762	223

The precision, recall and F1-measure metrics for each aspect:

Performance	F "+"	F "-"	F mean	Accuracy
Room	0.743	0.257	0.5	0.618
Location	0.871	0.076	0.473	0.773
Service	0.811	0.19	0.501	0.693

Dataset and code available at:
https://goo.gl/DTEpxs

Automatic Morphemic Analysis of Russian Words

Lyudmila Maltina (lpmaltina@gmail.com),
Alexey Malafeev (amalafeev@yandex.ru)

National Research University Higher School of Economics
Nizhny Novgorod

1. TASKS AND MATERIAL

Tasks

- Morphemic segmentation of words and texts
- Search of morphs

Material

the morpheme and spelling dictionary by A.N. Tikhonov + the 1980 Russian grammar

morph database with frequency and position data (17,017 different morphs)

gold standard word analyses used for testing (500 words, not used in the training data)

https://vnutrislova.net – the system, with which we compare the performance of our models

2. DEVELOPED MODELS

Rule-based:
rules;
rules_corrected

Probabilistic:
maxmatch;
log_likelihood;
mean

Combined:
rules_corrected + maxmatch,
rules_corrected + log_likelihood,
rules_corrected + mean

3. DESCRIPTION OF THE MODELS: Rule-based

Rules	*Rules_corrected*
The model considers: •the form-building patterns •derivational connections between words •the POS and other morphological features of the word	Has more accurate marking of prefixes compared to the model ***rules***

3. DESCRIPTION OF THE MODELS: Probabilistic

Maxmatch	*Log_likelihood*	*Mean*
A part of the word is considered a morph if it is included in the list of morphs and is the longest possible match	1) Select combinations of morpheme boundaries in which the resulting word segments can occur at a given position and are found in the list of morphs 2) Choose the candidate analysis with the maximum value of the logarithms	Compute the arithmetic mean of morph probabilities for each candidate analysis, choose the one with the greatest arithmetic mean

3. DESCRIPTION OF THE MODELS: Combined

rules_corrected + maxmatch, *rules_corrected + log_likelihood,* *rules_corrected + mean*
1) The *rules_corrected* model extracts postfixes, inflections, prefixes and suffixes 2) For finding the root and suffixes not found by *rules_corrected*, one of the three other models(*maxmatch*, *log_likelihood*, or *mean*) is used

4. EVALUATION

hits - the number of correct boundaries(true positives)
insertions - the number of unnecessary boundaries (false positives)
deletions - the number of overlooked boundaries (false negatives).

$$Precision = \frac{hits}{hits + insertions}$$

$$Recall = \frac{hits}{hits + deletions}$$

$$F - measure = \frac{2 \times hits}{2 \times hits + insertions + deletions}$$

5. RESULTS

Algorithm	Precision	Recall	F-score
rules	0.905	0.639	0.749
rules_corrected	**0.944**	0.63	0.756
maxmatch	0.73	0.567	0.638
log_likelihood	0.73	0.567	0.638
mean	0.652	0.795	0.716
rules_corrected + maxmatch	0.846	0.85	**0.848**
rules_corrected + log_likelihood	0.847	0.847	0.847
rules_corrected + mean	0.551	**0.915**	0.687
External system (https://vnutrislova.net)	0.834	0.713	0.769

6. CONCLUSION AND FURTHER WORK

The best-performing models allow to analyze:

- previously unseen words
- complex words
- words in non-initial forms

Further work:

- paying more attention to word-formative suffixes
- improving the algorithm for analyzing complex words
- Implementing the search for related words in a text