Machine learning technology for correcting electronic medical texts in Russian

Student: Dmitry Pogrebnoy, J42332c

Supervisor: Sergey Kovalchuk, PhD

Electronic medical records

- There are many medical models based on patients' medical records
- The quality of models mostly depend on the quality of the source texts
- Patient data is a plain text and contains a lot of spelling errors
- Spelling errors greatly reduce the quality of the final models
- Fixing such errors will improve the quality of the medical models

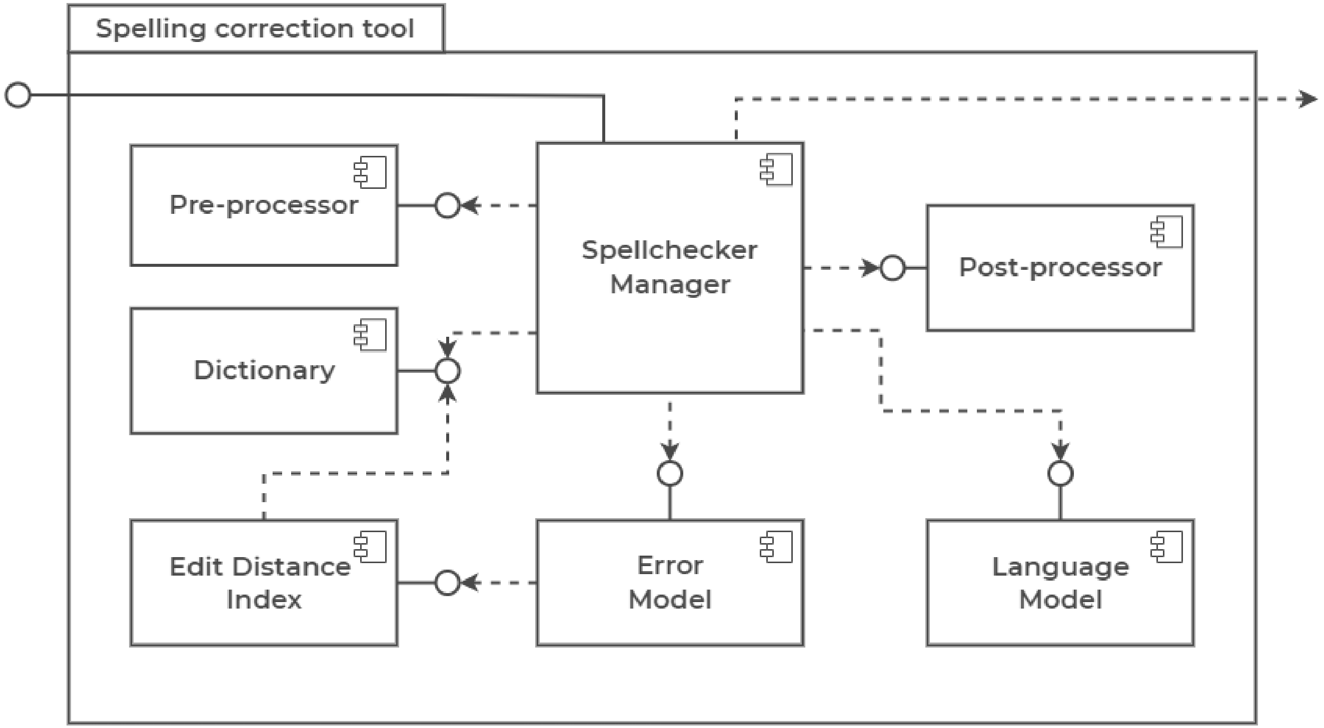
Goal and tasks

Goal: Design a method and implement an automatic spelling correction tool for clinical texts in Russian.

Tasks for third semester:

- Collect and prepare data for training language models
- Select and fine-tune BERT models for ranking task
- Conduct extensive testing of the developed tool
- Assemble the tool into a package and publish it

Tool architecture



Anamneses dataset

- Public datasets
  - RuMedNLI – 14716 records
  - RuMedPrimeData – 15249 records
- Private datasets
  - Almazov National Medical Research Center – 2355 records
  - Research Institute of the Russian Academy of Sciences – 161 records
- All datasets were pre-processed and assembled into final one
  - Tokenization and lemmatization
  - Stop words filtering

Fine-tune BERT models

- sberbank-ai/ruRoberta-large → MedRuRobertaLarge
  - Size – 1.4 Gb
- distilbert-base-multilang-cased → MedDistilBertBaseRuCased
  - Converted from multilang to Russian model
  - Size - 217 Mb
- cointegrated/rubert-tiny2 → MedRuBertTiny2
  - Size – 117 Mb
- Train/test/eval datasets – 0.8/0.1/0.1
- All models are published on the Hugging Face repository

Word tests internals

- Single test - error and lexical precision
  - 100 test sample per each error type
- Context test - error and lexical precision
  - 100 test samples per each error type
  - 10 words in each sample
  - One of ten words is incorrect, other words are correct
  - Same incorrect words as in single test
- Performance
  - Laptop with Ubuntu 20.04
  - 24 GB RAM and Intel Core i5-10750H CPU @ 1.60GHz * 12

Single word test

<table>
<tr><th>Tool</th><th>Error precision</th><th>Lexical precision</th><th>Average precision</th><th>Average words per second</th></tr>
<tr><td>Aspell-python</td><td>0.86</td><td>0.859</td><td>0.859</td><td>283.7</td></tr>
<tr><td>PyHunspell</td><td>0.812</td><td>0.539</td><td>0.675</td><td>9.4</td></tr>
<tr><td>PyEnchant</td><td>0.829</td><td>0.541</td><td>0.685</td><td>20</td></tr>
<tr><td>LanguageTool-python</td><td>0.762</td><td>0.904</td><td>0.833</td><td>25.1</td></tr>
<tr><td>PySpellChecker</td><td>0.354</td><td>0.86</td><td>0.607</td><td>3.4</td></tr>
<tr><td>SymspellPy</td><td>0.399</td><td>0.813</td><td>0.606</td><td>9702.8</td></tr>
<tr><td>Jumspell</td><td>0.267</td><td>0.947</td><td>0.607</td><td>2552.1</td></tr>
<tr><td>Tool (CPU, MedDistilBert)</td><td rowspan="2">0.701</td><td rowspan="2">0.981</td><td rowspan="2">0.841</td><td>12.7</td></tr>
<tr><td>Tool (GPU, MedDistilBert)</td><td>39.7</td></tr>
</table>

Context word test

<table>
<tr><th>Tool</th><th>Error precision</th><th>Lexical precision</th><th>Average precision</th><th>Average words per second</th></tr>
<tr><td>Aspell-python</td><td>0.739</td><td>0.93</td><td>0.835</td><td>357.3</td></tr>
<tr><td>PyHunspell</td><td>0.706</td><td>0.719</td><td>0.713</td><td>11.8</td></tr>
<tr><td>PyEnchant</td><td>0.721</td><td>0.719</td><td>0.72</td><td>24.3</td></tr>
<tr><td>LanguageTool-python</td><td>0.727</td><td>0.942</td><td>0.835</td><td>43.6</td></tr>
<tr><td>PySpellChecker</td><td>0.304</td><td>0.868</td><td>0.586</td><td>6.7</td></tr>
<tr><td>SymspellPy</td><td>0.37</td><td>0.913</td><td>0.642</td><td>26060.2</td></tr>
<tr><td>Jumspell</td><td>0.307</td><td>0.969</td><td>0.638</td><td>4322.3</td></tr>
<tr><td>Tool (CPU, MedDistilBert)</td><td rowspan="2">0.734</td><td rowspan="2">0.984</td><td rowspan="2">0.861</td><td>45.47</td></tr>
<tr><td>Tool (GPU, MedDistilBert)</td><td>134.453</td></tr>
</table>

Python package

- Assembled the pip python package
- Package contains
  - Source code
  - Dictionary with correct words
  - No models included
- Models are loaded dynamically as needed
- Published package name – medspellchecker

Conclusion

- Dataset for training language models is collected
- Three different BERT models are fine-tuned for ranking task
- Extensive testing of the developed tool is conducted
- Package with new tool is assembled

Fine-tuned models
huggingface.co/DmitryPogrebnoy

Pip package
pypi.org/project/medspellchecker

Further plans

- Improve and optimize the spelling correction process
- Try to fine-tune smaller language models and test them
- Evaluate the effect of the developed tool on medical models

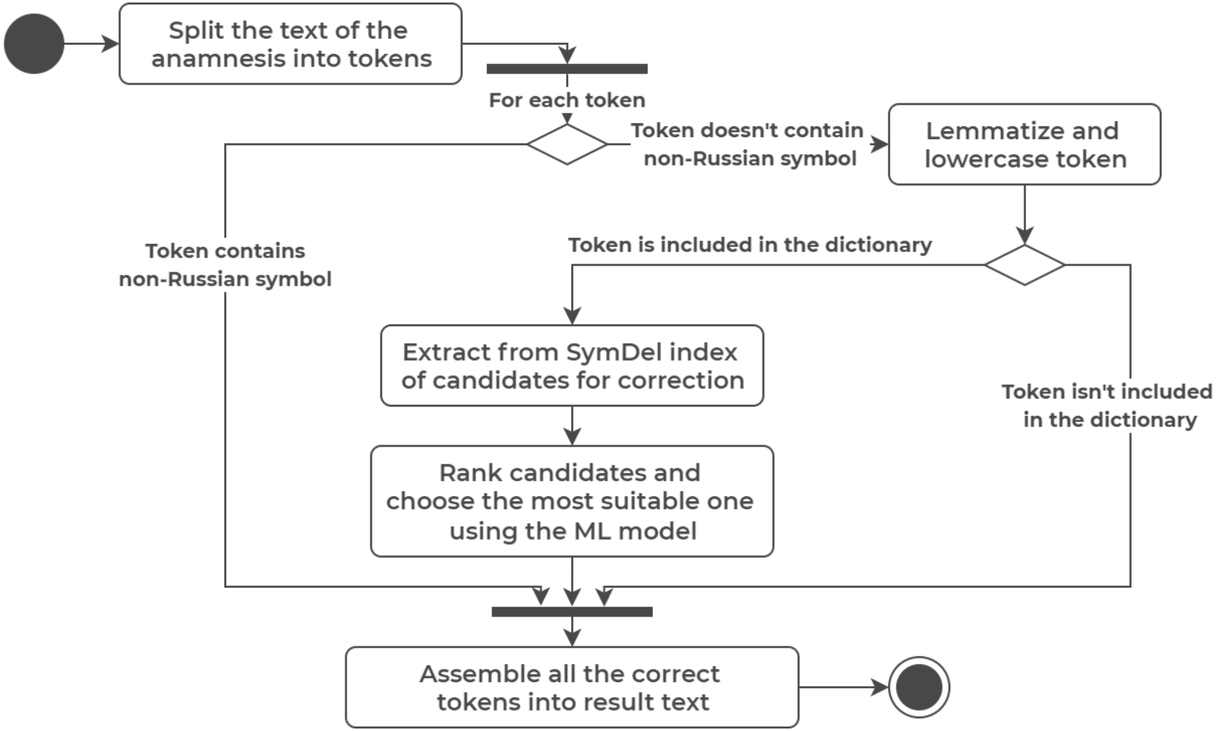
Thank you for your attention!

ITsMOre than a UNIVERSITY

Spelling errors in question

Type of mistake	Incorrect text	Correct text
Wrong characters	тубиркулез	туберкулез
Missing characters	туб□ркулез	туберкулез
Extra characters	туберкплулез	туберкулез
Shuffled characters	тубрекулез	туберкулез
Missing word separator	острый|туберкулез	острый_туберкулез
Extra word separator	туб_еркулез	туберкулез

Spelling correction process



Metrics

- **Error precision** – the ratio of the number of correctly corrected words to the total number of incorrect words
- **Lexical precision** – the ratio of the number of unchanged modified words to the total number of correct words
- **Average precision** – the average of error precision and lexical precision
- **Performance** – the number of words processed by the tool per second