3rd MLFPM Summer School

A practical guide to information extraction from medical texts

Sven Laur
University of Tartu

https://github.com/swenlaur/information-extraction-tutorial

Why information extraction is needed

Electronic health records are mostly unstructured:

- ◇ patient complaints (adverse drug reactions)
- ◇ disease descriptions are textual (deep phenotyping)
- ◇ biopsies have textual descriptions (cancer studies)
- ◇ descriptions of X-ray scans are textual (label assignment)

Information extraction allows us:

- ◇ to fill gaps in the structured data (allergies)
- ◇ to describe environment factors (lifestyle and family history)
- ◇ to refine diagnosis description (infraction subtypes)
- ◇ to refined treatment outcomes (stroke complications)

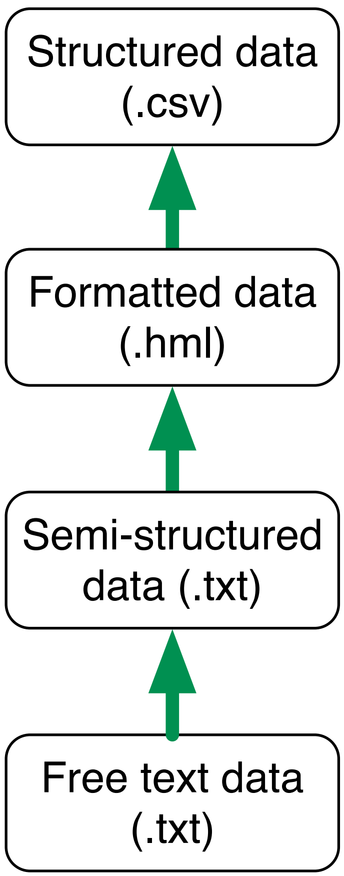
Example. Measurement extraction

Extract dated lab measurements from a patient health record.

unit 1 | value 2 | analyte 3 | date 4

14.10.2016 16:43 date: Kolesterool analyte 3.5 value mmol/L unit [norm ... - 5.0]
Ametikoht ja staaž: lukksepp / 20 aastat
Analüüsid: LK analyte 7,7; value valem normis; Hgb analyte 103 value g/l unit;
ER analyte 3,6 value; MCV analyte 89,6 value; T 420 value; transaminaasid normis.
EKG reg. siinusrytm 75 l.min horis s el posits.
Jätkub TVL, uus kontakt 02.03.18 date.
Kaebused halvale enesetundele ja säärelihastes krampidele.
Kaela paremal poolel puuk.
Kliiniline veri: haigusliku leiuta; ALAT analyte 32 value U/l unit (norm), ASAT analyte
26 value U/l unit (norm)
Külmatunne pea surin, jalalabad surisevad Ei talu vett, on füüsiliselt valus

NLP tasks in medical domain



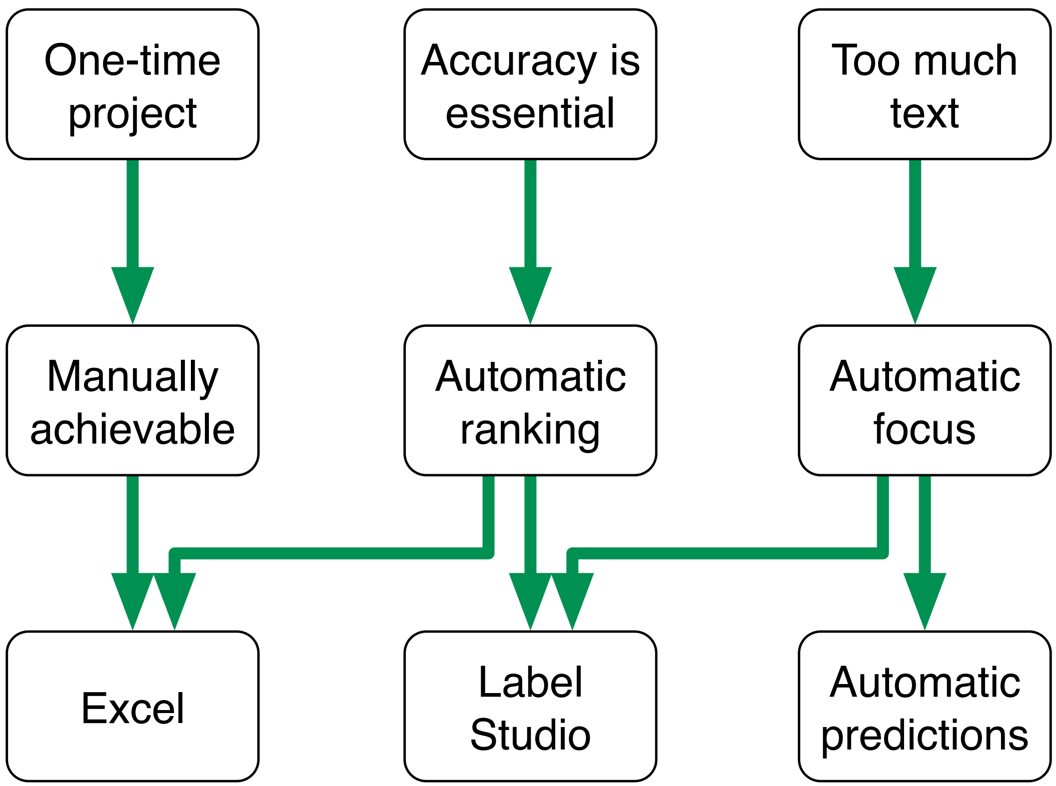
- Cleaning individual values (1, 3 $\rightsquigarrow$ 1.3)
- Standardisation of terms (penicillin $\rightsquigarrow$ J01C)
- Harmonisation and conflict resolution

- Extraction of individual data fields
- Unification of different data formats

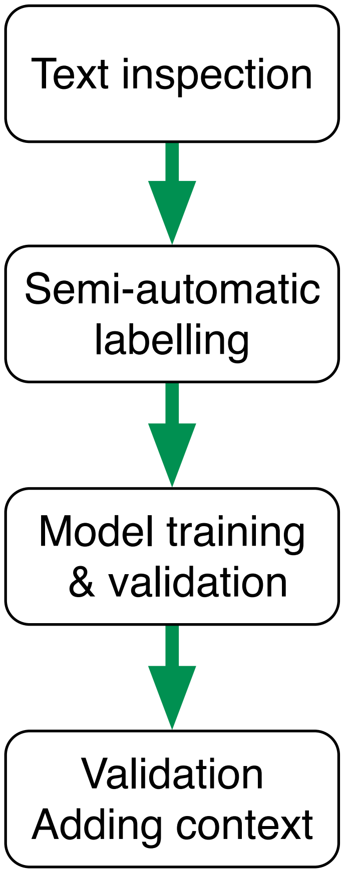
- Format discovery
- Robust parsing
- Text segmentation

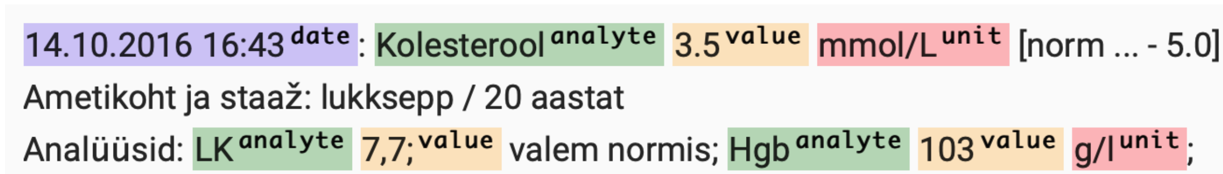
- Text segmentation
- Relevance ranking
- Fact extraction

How to carry out information extraction



Typical steps for a fact extraction task



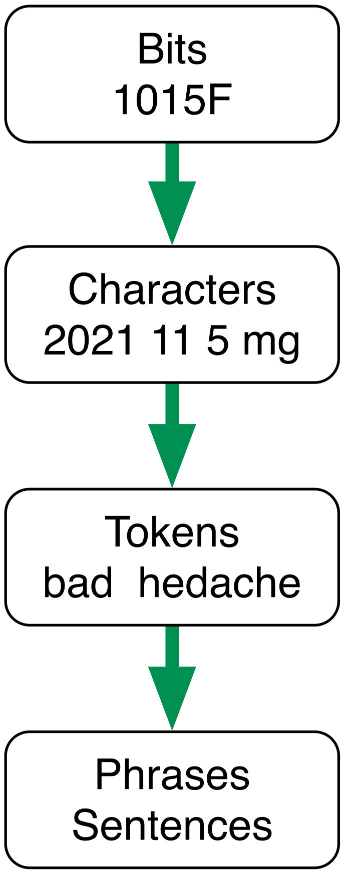


- ▷ Term lists and regular expressions
- ▷ Text segmentation and focus regions
- ▷ Dataset enrichment. Manual corrections

- ▷ Detection and extraction tasks
- ▷ Text prioritisation task

- ▷ Plausibility (Weight: 1.63 kg)
- ▷ Context (Weight: 3.4 kg 30 year female)

Abstraction levels in text mining



- ▷ Charset detection (utf-8, Latin-1, Windows-1257)
- ▷ Recovery from encoding errors (jΓjriΓ¶Γ¶ ⇝ jüriöö)

Tokenisation
- ▷ number and abbrevation detection (weight 1, 3 kg)
- ▷ word normalisation (hedache⇝headache, ug⇝ μg)

Token-level annotations
- ▷ morphological analysis (cramped ⇝ verb, past tense)
- ▷ term ontologies (penicillin ⇝ J01C, liver ⇝ abdomen)

Phrase level analysis
- ▷ text segmentation and relevance ranking (focus)
- ▷ fact extraction and text quantification (prediction)

Commonly used methods

Knowledge based → Supervised machine learning → Unsupervised machine learning

Rule-based methods

- ▷ term lists
- ▷ regular expressions & rewriters
- ▷ phrase grammars

Lexicons
Ontologies
Standards

Supervised machine learning

- ▷ text segmentation
- ▷ text classification
- ▷ keyword assignment

Text annotations

CPU & GPU time

Unsupervised machine learning

- ▷ word embeddings (WORD2VEC)
- ▷ transformers (BERT & GPT-3)
- ▷ similarity scoring (WMD)

Unlabelled text

Few text annotations

CPU & GPU time

Rule-based methods

Standards as the source of term lists

Disease names

◇ ICD-10, SNOMED-CT

Lab measurements

◇ LOINC, SNOMED-CT

Drug names and adverse reactions

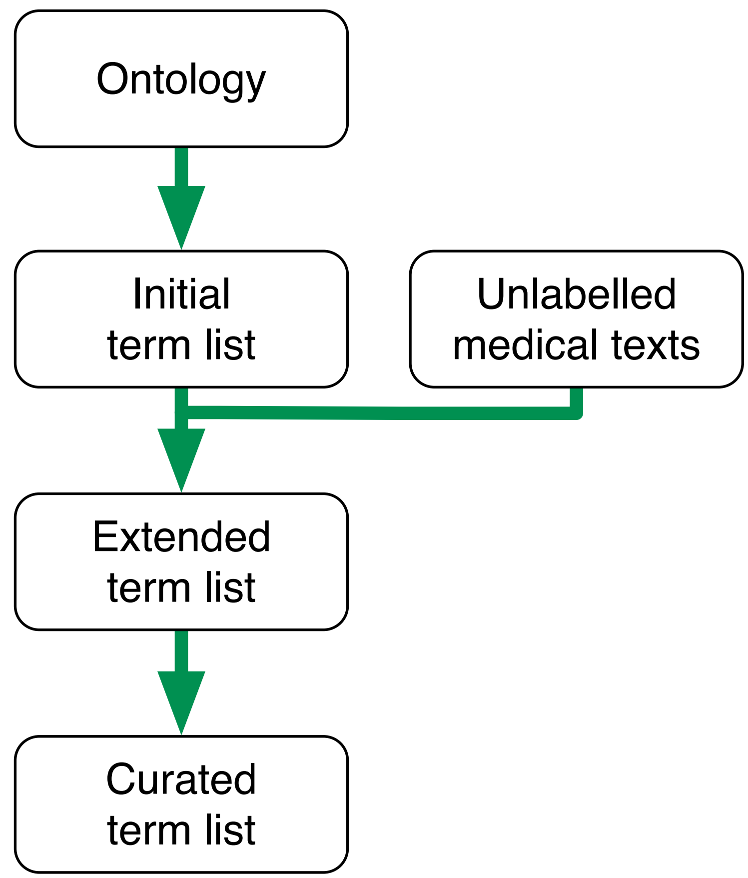
◇ ATC, SNOMED-CT

◇ drug package leaflets

Anatomy

◇ AEO, CARO, SNOMED-CT

◇ medical dictionaries



Tokenisation with regular expressions

Regular expression is a way to specify tokens with fixed structure:

- dates (`[0-9]^2.[0-9]^2.[0-9]^4`)
- number formats (`-?[0-9]^+.[0-9]^+`)
- special symbols, headers

Do **not** use regular expression for predictable variations in term lists.

- handling long term lists is much more efficient.
- you might get `a|ab|abc` vs `abc|ab|a` problems.

Software development practices:

- Write test cases for each surprise.
- Maintain common library of regular expressions.

What is a phrase grammar

A phrase grammar is a list of rules that combines tokens into phrases:

$$\begin{aligned}
\textsc{Number Unit} &\rightsquigarrow \textsc{QNumber}\\
\textsc{Date Analyte QNumber} &\rightsquigarrow \textsc{Measurement}\\
\textsc{Analyte QNumber} &\rightsquigarrow \textsc{Measurement}\\
\textsc{Date Analyte Number} &\rightsquigarrow \textsc{Measurement}\\
\textsc{Analyte Number} &\rightsquigarrow \textsc{Measurement}
\end{aligned}$$

- ▷ There can be several phrase symbols of interest.
- ▷ If many rules apply the one with highest priority is applied.
- ▷ Rules can specify how to compute extra attributes for derived symbols.
- ▷ All phrase grammars classes are finite in practice.

Illustrative example

Canonical phrase

`21.05.2021` `Cholesterol` `100.2` `mg/dL`

$$\rightsquigarrow$$

$$\textsc{Measurement}(date = 2021/05/21, analyte = Cholesterol, \ldots)$$

Incompleted phrases we must match

- `Cholesterol` `100.2` `mg/dL`
- `21.05.2021` `Cholesterol` `100.2`
- `Cholesterol` `100.2`

Advanced tricks

Incomplete phrases reveal missing knowledge

- `21.05.2021` `HDL` `20.3` `mg/dL`
- `21.05.2021` `Cholesterol` `406` `mg/L`

Additional checks

- `Last measurement of` `Cholesterol` `2005`

Tokenisation is ambiguous in practice

- `Cholesterol` `21.05 2021` `40.6` `;` `HDL`
- `Cholesterol` `21.05` `2021` `40.6 ;` `HDL`

Conclusion

Advantages of rule-based methods:

- ▷ Easy to achieve decent progress.
- ▷ Do not need extensive manual annotations.
- ▷ A good baseline for segmentation tasks.

Disadvantages of rule-based methods:

- ▷ Manual derivation of rules is hard.
- ▷ Curation of background information is hard.
- ▷ Progressively harder to improve the performance.

Supervised machine learning

Support Vector Machines

Linear classifier is an automatic way to derive implicit rules from examples

$$f(x) = 1 \times \boxed{\texttt{Cholesterol}}(x) + 1 \times \boxed{\texttt{HDL}}(x) + 1 \times \boxed{\texttt{LDL}}(x) - 2$$

$$\Updownarrow$$

$$\boxed{\texttt{Cholesterol}}(x) \wedge \boxed{\texttt{HDL}}(x) = \textsc{True}$$

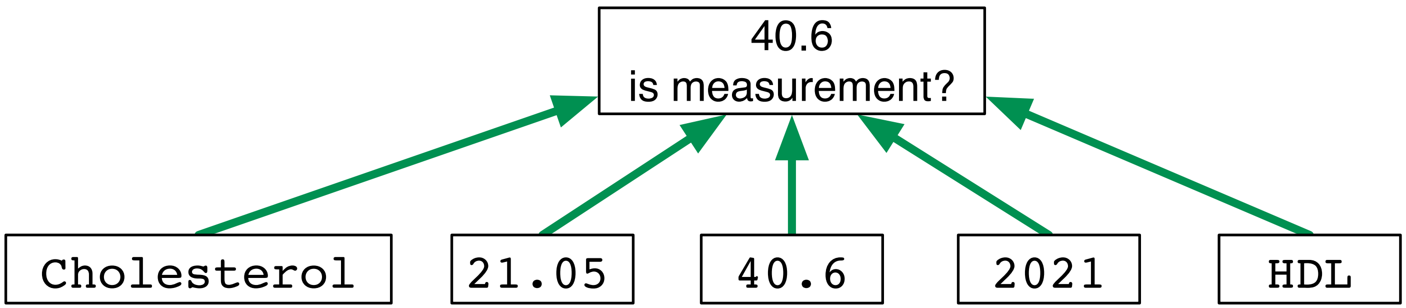
$$\boxed{\texttt{HDL}}(x) \wedge \boxed{\texttt{LDL}}(x) = \textsc{True}$$

$$\boxed{\texttt{Cholesterol}}(x) \wedge \boxed{\texttt{LDL}}(x) = \textsc{True}$$

Support Vector Machine is *statistically stable way* to do linear classification.

▷ Feature maps and kernels allow to do nonlinear combination of features.

Manual feature engineering



The quality of predictions mostly depends on available features:

- ▷ term lists
- ▷ phrase lists
- ▷ morphological features
- ▷ size of the window

Output smoothing with CRF

- The predictions of SVM are independent for each position.

  `21.05` 2021 `Cholesterol` `40.6` ; `HDL` `16.3` `mg/dL`

- Consistency requirements can be modelled with Markov random fields.

  very unlikely

- Conditional Random Fields smooths independent predictions.

  `21.05` 2021 `Cholesterol` `40.6` ; `HDL` `16.3` `mg/dL`

  `21.05` `2021` `Cholesterol` `40.6` ; `HDL` `16.3` `mg/dL`

Word embeddings

Word embedding defines 100-1000 informative features for each word.

- ▷ Features are defined automatically.
- ▷ Masked language modelling is used for training.
- ▷ Each word gets a fixed representation vector.
- ▷ Cosine similarity between embeddings indicates semantical closeness.

By running SVM on top of embeddings:

- ▷ We do not need to hand-crafting word-based features.
- ▷ We still have to think about the window size.
- ▷ We have to fix the amount of unknown words.
- ▷ We ignore that words can have multiple meanings.

Context sensitive word embeddings

Neural networks define informative features for each occurrence of the word.

- ▷ Features are defined automatically.
- ▷ Masked language modelling is used for training.
- ▷ There is no observation window – information can flow.
- ▷ Different meanings of the words get different representations.
- ▷ Sentences or paragraphs get also representations.

By running a neural network on top of context-sensitive embeddings:

- ▷ We can adjust the baseline representation for current task.
- ▷ We still cannot capture dark background knowledge.

Iterative improvement

Three main sources of improvement

Improve the quality of term lists and ontologies.

- ▷ Use version control smartly to communicate changes

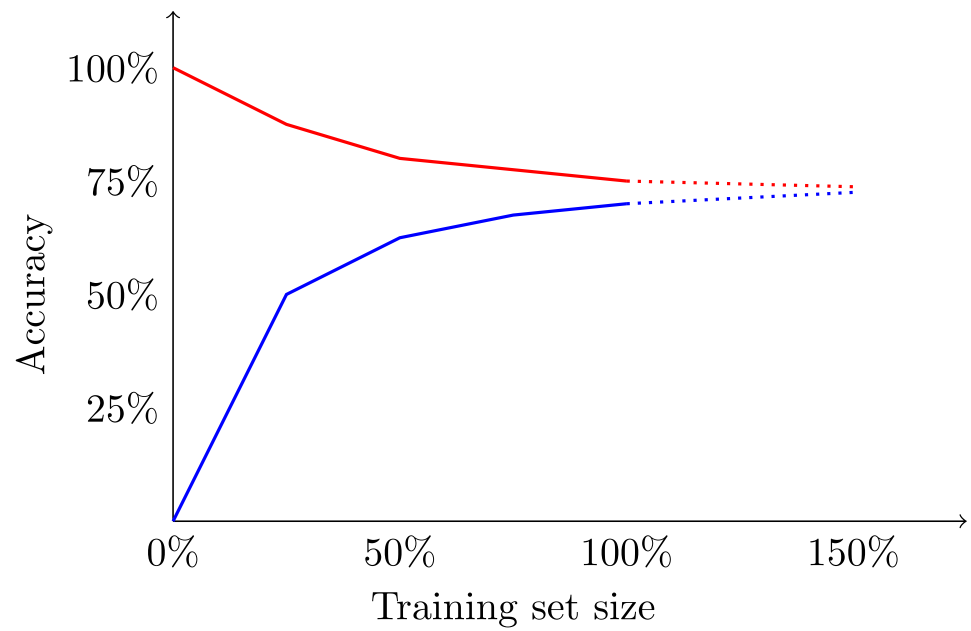
Extraction quality does not increase if you do not measure it!

- ▷ Create dedicated test sets for each isolated problem.

Use external consistency checks to discover errors.

- ▷ Any test that works unlabelled data is good.
- ▷ Automatic checks for statistical anomalies.

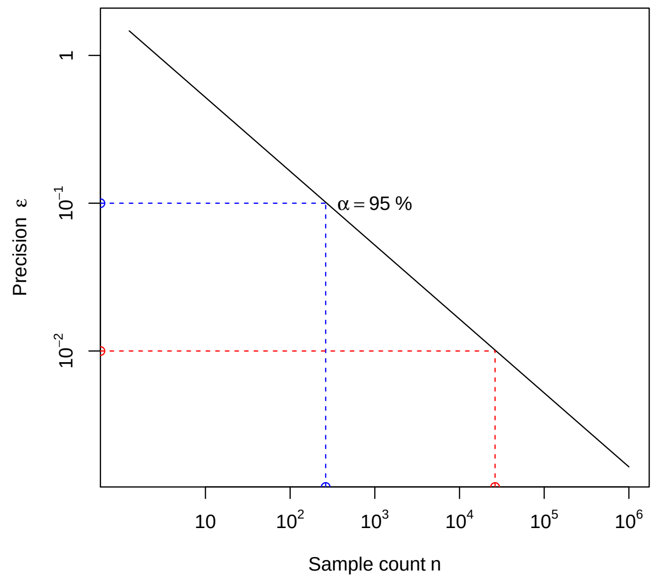
Diminishing returns



Most machine learning problems are not solvable by collection more samples.

▷ By reducing training set size it one can estimate potential gains.

Pitfalls of absolute performance measures



Test error estimates are not very precise:

- ▷ To increase precision 10 time you need 100 times more data.
- ▷ You can estimate test error with precision 1% not more.
- ▷ You cannot reliably detect progress on a reasonable test set.

Relative performance

Unlabelled data can be used for more precise performance estimates:

- ▷ Fix a good base line model.
- ▷ Evaluate both models on unlabelled data.
- ▷ Choose uniformly at random 100 - 1000 prediction differences.
- ▷ Establish the ground truth for these differences.
- ▷ Compute improvement ratio for differences.
- ▷ Compute relative frequency of difference.
- ▷ Their multiple is the relative performance gain.