Accessing the Republic.
Entity extraction from the resolutions of the Dutch States-General

Marijn Koolen,[1,2] Esger Renkema,[1,2] Nienke Groskamp,[1] Frank Smit,[1]
Jirsi Reinders,[1] Ronald Sluijter,[1] Rik Hoekstra,[1,2] Jorid Oddens[1]

[1]Huygens Institute, Amsterdam, Netherlands
[2]DHLab - KNAW Humanities Cluster, Amsterdam Netherlands

DH Benelux 2024 - 6 June 2024 - Leuven, Belgium

REPUBLIC

huygens instituut

- REsolutions PUBLished In a Computational environment
  - Resolutions of the States General of the Dutch Republic (1576-1796)
  - Long serial publication (220 years), over 500,000 pages
  - ~60,000 daily meeting, ~1 million resolutions (propositions and decisions)

Motivation

- Reasons to tag entities
  - Additional access points: alternative paths to navigate between documents
  - Contextual information: quick assessment of relevance/interest for user
- Reasons not to tag entities
  - If quality is low, it can annoy users, induce a lack of trust
  - Added value may not outweigh required effort
- Which entities?
  - Which entities occur?
  - Which are interesting?
  - Which are taggable?

Entities in the Resolutions

- Tagging 8 types of entities
  - Person: person name including any attributions (title, job, legal status, ...)
  - Attribution: person attribution (title, job, legal status, ...) if refers to specific entity
  - Organisation: any organisation (incl. region name when it refers to governing body)
  - Committee: members of the States General tasked to investigate a matter
  - (Geo)Location - Political entity: name of a geolocation when it refers to the place
  - Date: absolute and relative dates (of submitted propositions and previous resolutions)
  - Resolution reference: references to specific earlier resolutions
  - Other: any remaining names

Tagging Project

- Ground Truth
  - 1631 tagged resolutions of printed volumes 1705-1796
    - 370,560 tokens, 23,875 entities
  - 513 tagged paragraphs of handwritten volumes 1597-1702
    - 28,387 tokens, 2,347 entities
- Automated tagging
  - Train and evaluate taggers per entity type

Nested Entities and Ground Truth

- Entities can be highly complex, with multiple levels of nesting
  - E.g. person name + attribution
  - Attribution can contain an organisation which can contain another organisation which can contain a location
- Examples
  - Henricus Gerhardus de Beveren Esveld, Predikant in de Gereformeerde Gemeente te Schoondyke onder het Classis van Walcheren, be roepen zynde tot Predikant in de Gemeente te Enkhuisen
  - Jan van Reusen — Solliciteur van den heer Thibaut heer van St. Aechtekercke, Burgermeester der Stadt middelburch cum socijs, taeckende de Dijckagie genaempt de Polder benoorden Aerdenburch

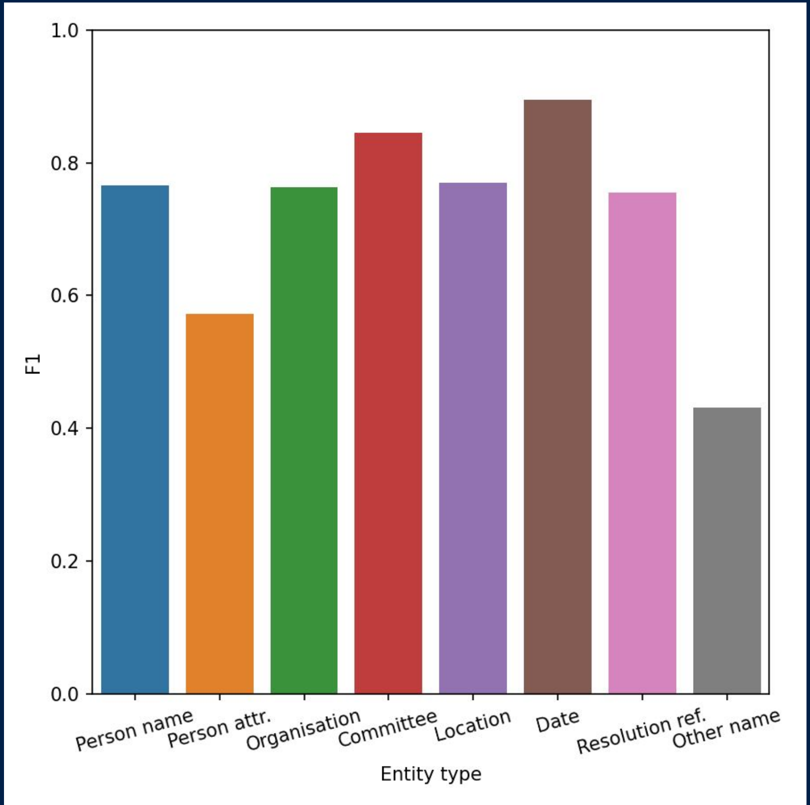
Training NER Taggers

- Combine/compare types of embeddings
  - Character-level embeddings, trained on resolution corpus (150 million words)
  - FastText embeddings, trained on resolution corpus
  - GysBERT (Manjavacas & Fonteyn 2022)
- Use Python Flair package (Akbik et al. 2019)
  - https://flairnlp.github.io/flair/
  - Agnostic to type of embeddings: Character-, word-, sentence-level
  - Combine via stacked embeddings!

Combined Model or Model per Type?

- Single model
  - Advantages:
    - Need to train only one model, no tag conflicts in applying
  - Disadvantages:
    - Model choices may not be optimal for all entity types
    - Difficult to determine hierarchy of nested entities (ground truth is flattened)
- One per type
  - Advantages:
    - pick optimal model per type, allow for partial overlapping entities
    - Ground truth contains all information
  - Disadvantages:
    - Need decide how to deal with tag conflicts (partial overlap)

Quantitative Evaluation - Best Model Per Type



Evaluation

huygens instituut

Entity type	Tag repr.	Embeddings	Prec.	Recall	F1	Support
Person	single type	GysBERT + Char.	0.81	0.69	0.75	405
Person attr.	single type	GysBERT + FastText + Char.	0.57	0.56	0.56	573
Organisation	single type	GysBERT + FastText + Char.	0.82	0.71	0.76	283
Committee	single type	Char.	1.00	0.73	0.85	41
Location	single type	GysBERT + FastText	0.79	0.76	0.77	570
Date	single type	Char.	0.90	0.88	0.89	249
Resolution ref.	all types	GysBERT + FastText	0.82	0.70	0.75	57
Other names	single type	GysBERT + FastText + Char.	0.63	0.26	0.36	47

All best models use RNN instead of linear layer, CRF for prediction to capture dependencies in outputs

Tagging All Resolutions

- 1.5 million paragraphs
- 8 million entities
- Mostly persons and attributions

Entity type	# distinct	# total
Person	1,159,672	1,929,235
Person attribution	1,176,039	1,743,086
Organisation	287,022	743,860
Committee	70,518	135,198
Location	617,542	2,551,180
Date	411,477	873,202
Resolution reference	28,990	189,865
Other names	~	~
Total	3,751,260	8,165,626

Data Curation of tagged entities

- Identification of entities to recognise
- Resolution of entity descriptions to recognised entities

Data Curation of tagged entities

thousands!

very many!

- Identification of entities to recognise

millions!

- Resolution of entity descriptions to recognised entities

Where and how

to apply manual input?

Sources of variation

- multiple names / lack of formulaicity
- political developments / evolution through time
- inconsistent spelling / abbreviations
- text recognition errors

Sources of ambiguity

- multiplicity of names / descriptions
- intra-textual references

  „the matter in question"
  „aforementioned town"

Sources of variation

- multiple names / lack of formulaicity
- political developments / evolution through time

manual intervention

- inconsistent spelling / abbreviations
- text recognition errors

fuzzy matching, automatic re-writing

Sources of ambiguity

- multiplicity of names / descriptions
- intra-textual references

„the matter in question"
„aforementioned town"

may be resolved by examining context

Data curation:

Traditional model

Sources of variation

- multiple names / lack of formulaicity
- political developments / evolution through time

manual intervention

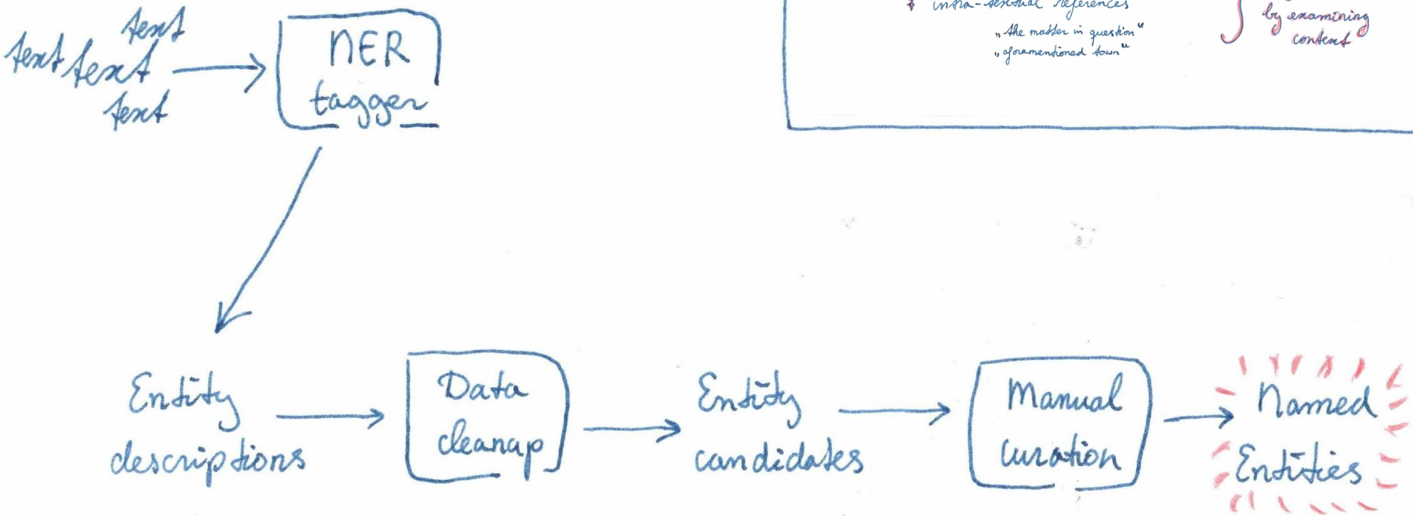
- inconsistent spelling / abbreviations
- text recognition errors

fuzzy matching, automatic re-writing

Sources of ambiguity

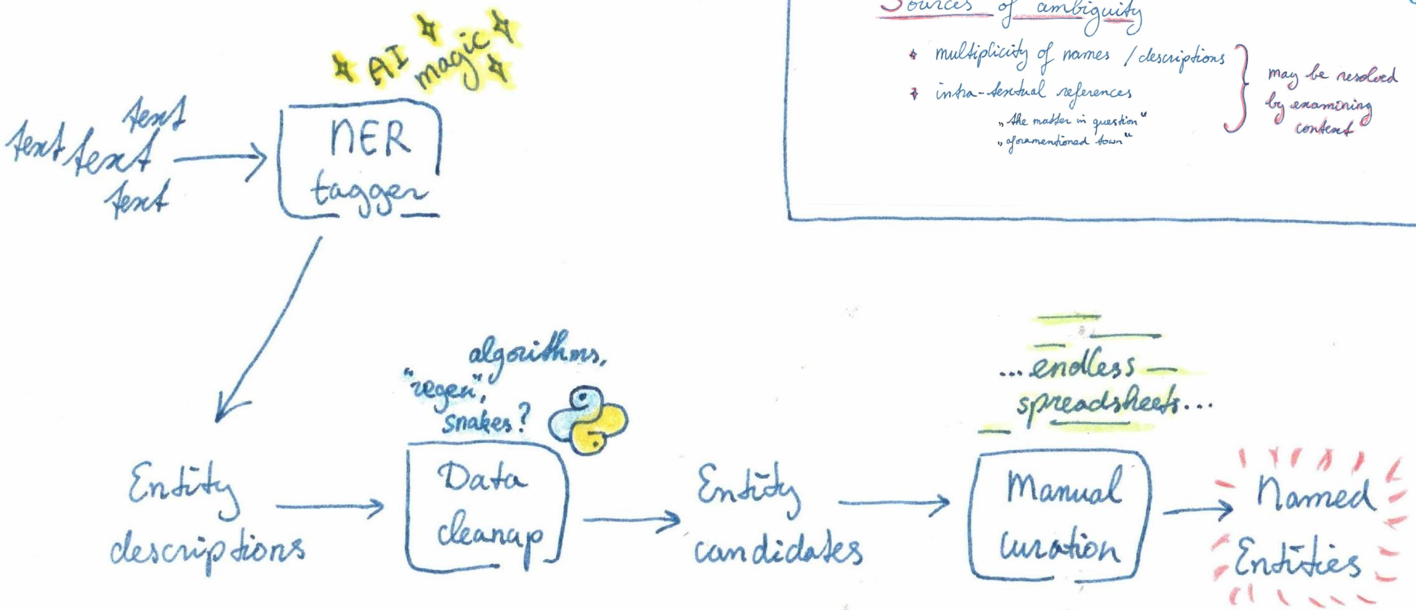
- multiplicity of names / descriptions
- intra-textual references
  „the matter in question“
  „aforementioned town“

may be resolved by examining context



Data curation:

Traditional model



Sources of variation

- multiple names / lack of formulaicity
- political developments / evolution through time

} manual intervention

- inconsistent spelling / abbreviations
- text recognition errors

} fuzzy matching, automatic re-writing

Sources of ambiguity

- multiplicity of names / descriptions
- intra-textual references
  „the matter in question"
  „aforementioned town"

} may be resolved by examining context

Data curation:

Traditional model

Sources of variation

- multiple names / lack of formulaicity
- political developments / evolution through time

} manual intervention

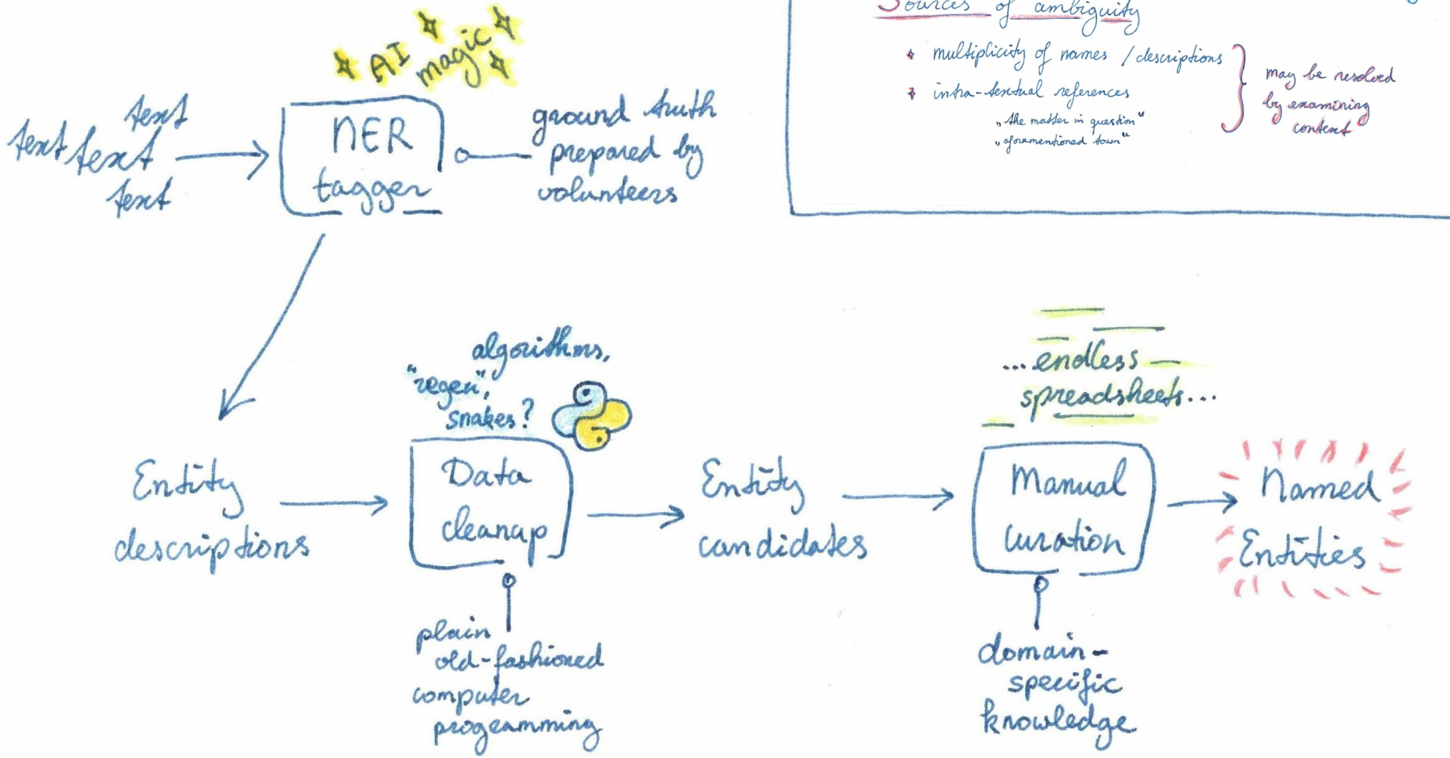
- inconsistent spelling / abbreviations
- text recognition errors

} fuzzy matching, automatic re-writing

Sources of ambiguity

- multiplicity of names / descriptions
- intra-textual references
  „the matter in question"
  „aforementioned town"

} may be resolved by examining context



Data curation:

Traditional model

Sources of variation

- multiple names / lack of formulaicity
- political developments / evolution through time

} manual intervention

- inconsistent spelling / abbreviations
- text recognition errors

} fuzzy matching, automatic re-writing

Sources of ambiguity

- multiplicity of names / descriptions
- intra-textual references
  „the matter in question"
  „aforementioned town"

} may be resolved by examining context

✦ AI magic ✦

text text text text → NER tagger

ground truth prepared by volunteers

must be content-agnostic

algorithms, "regex", snakes?

Entity descriptions → Data cleanup → Entity candidates → Manual curation → Named Entities

limited context dependence

plain old-fashioned computer programming

only practical on small data sets

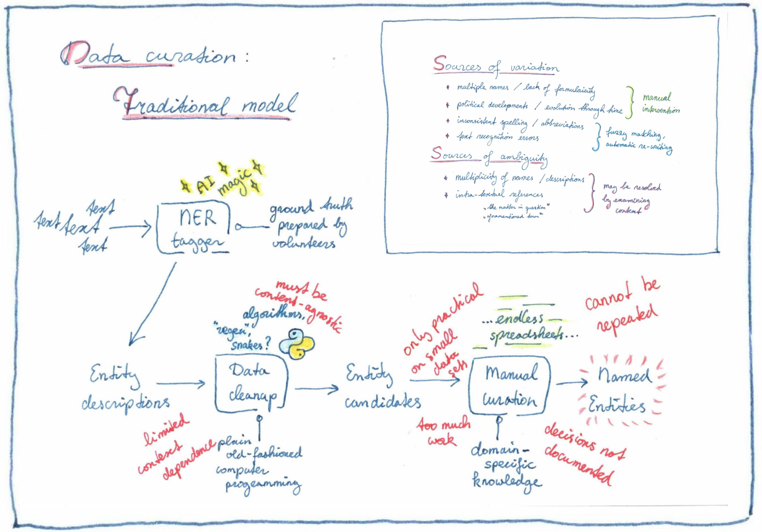
... endless spreadsheets ...

cannot be repeated

too much work

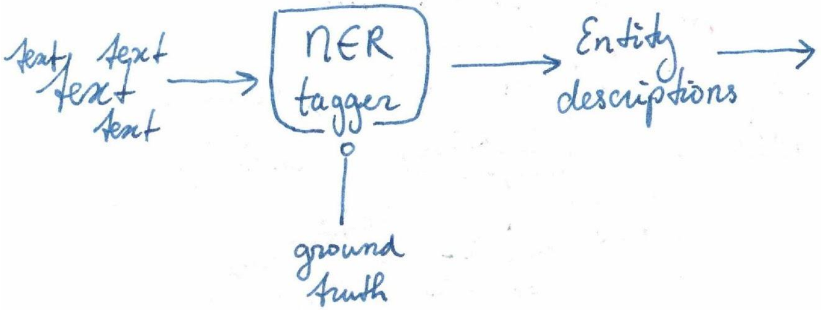
domain-specific knowledge

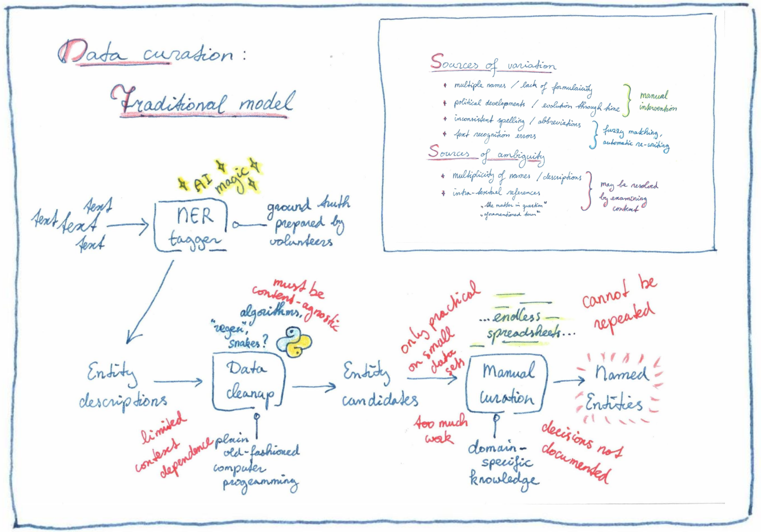
decisions not documented



Data curation:

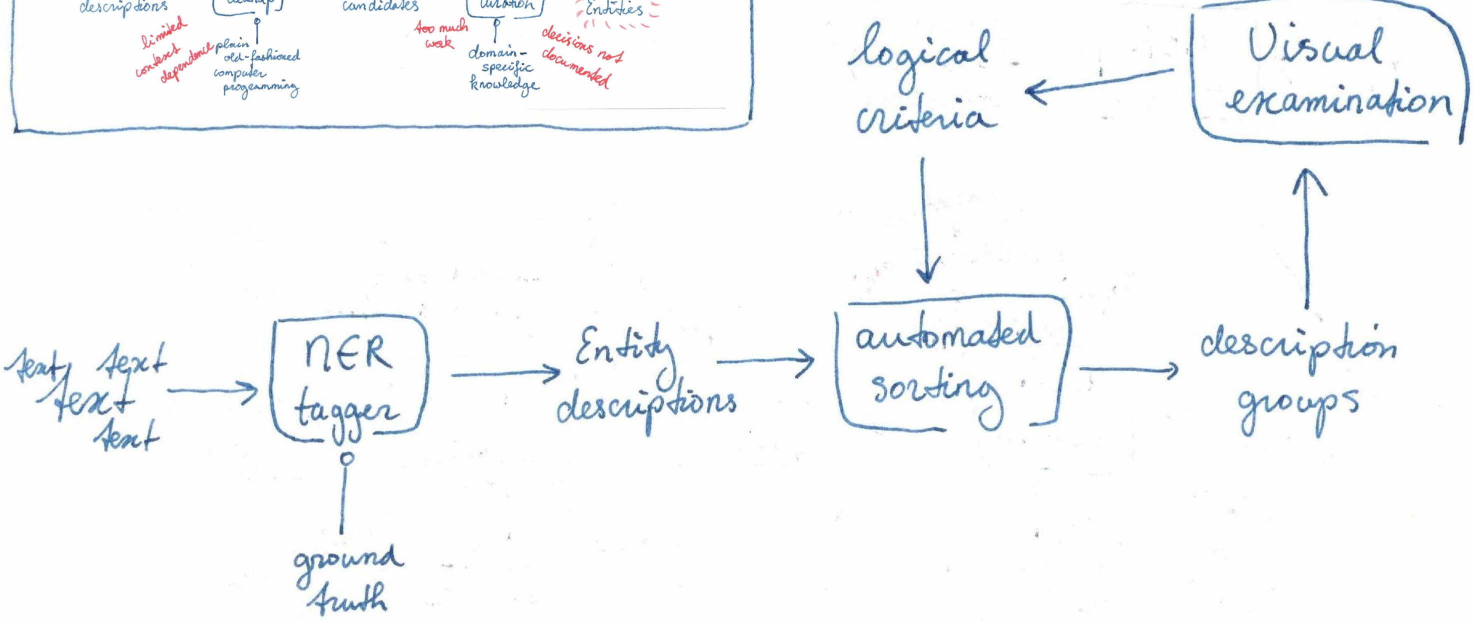
Integrated model





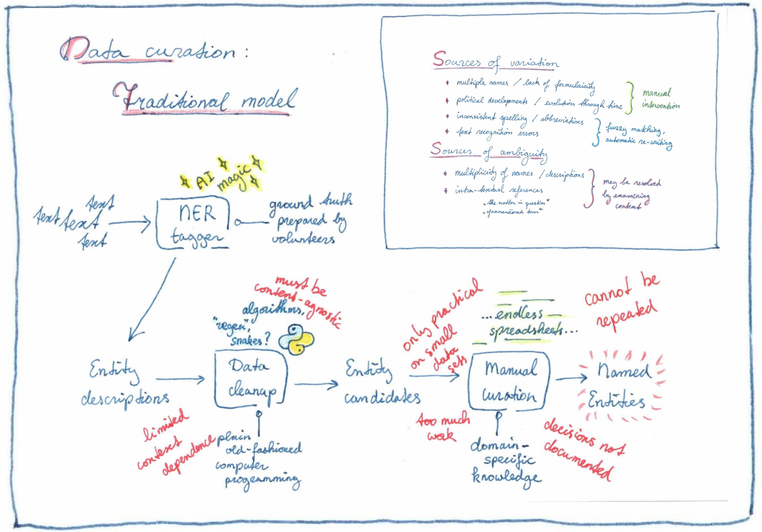
Data curation:

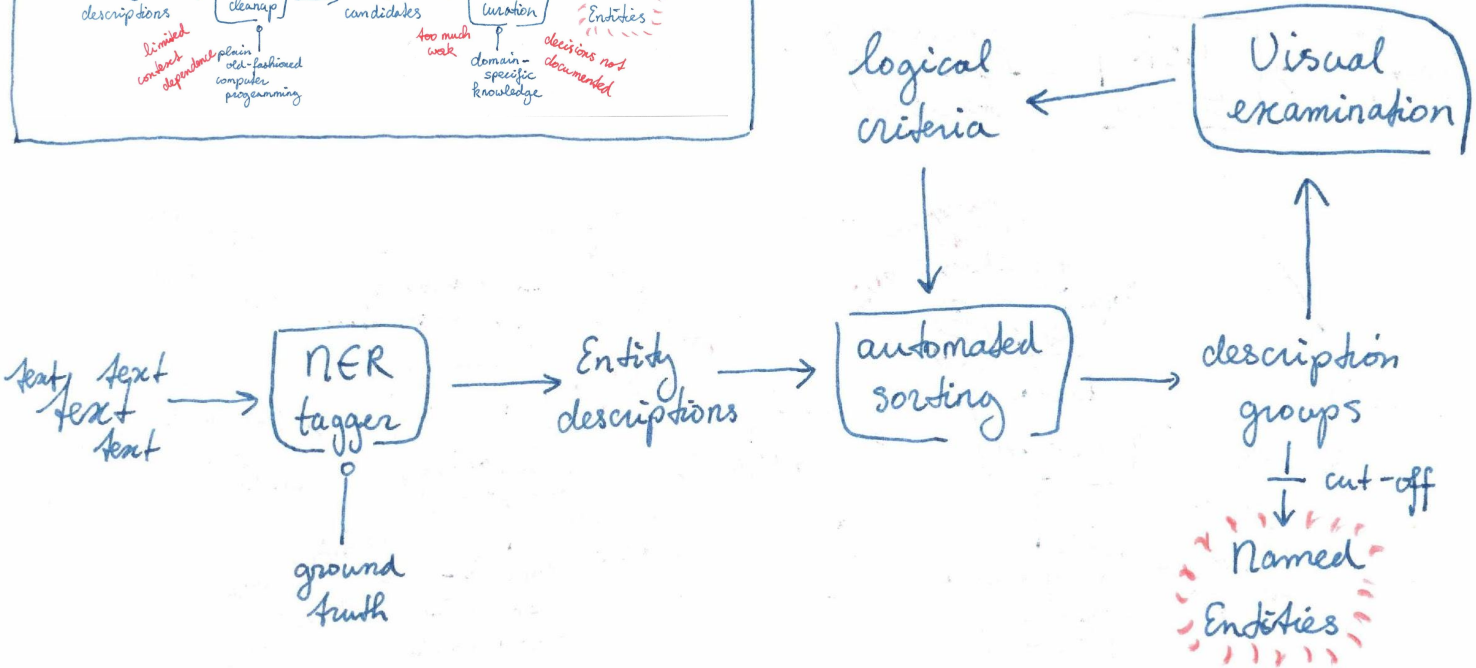
Integrated model

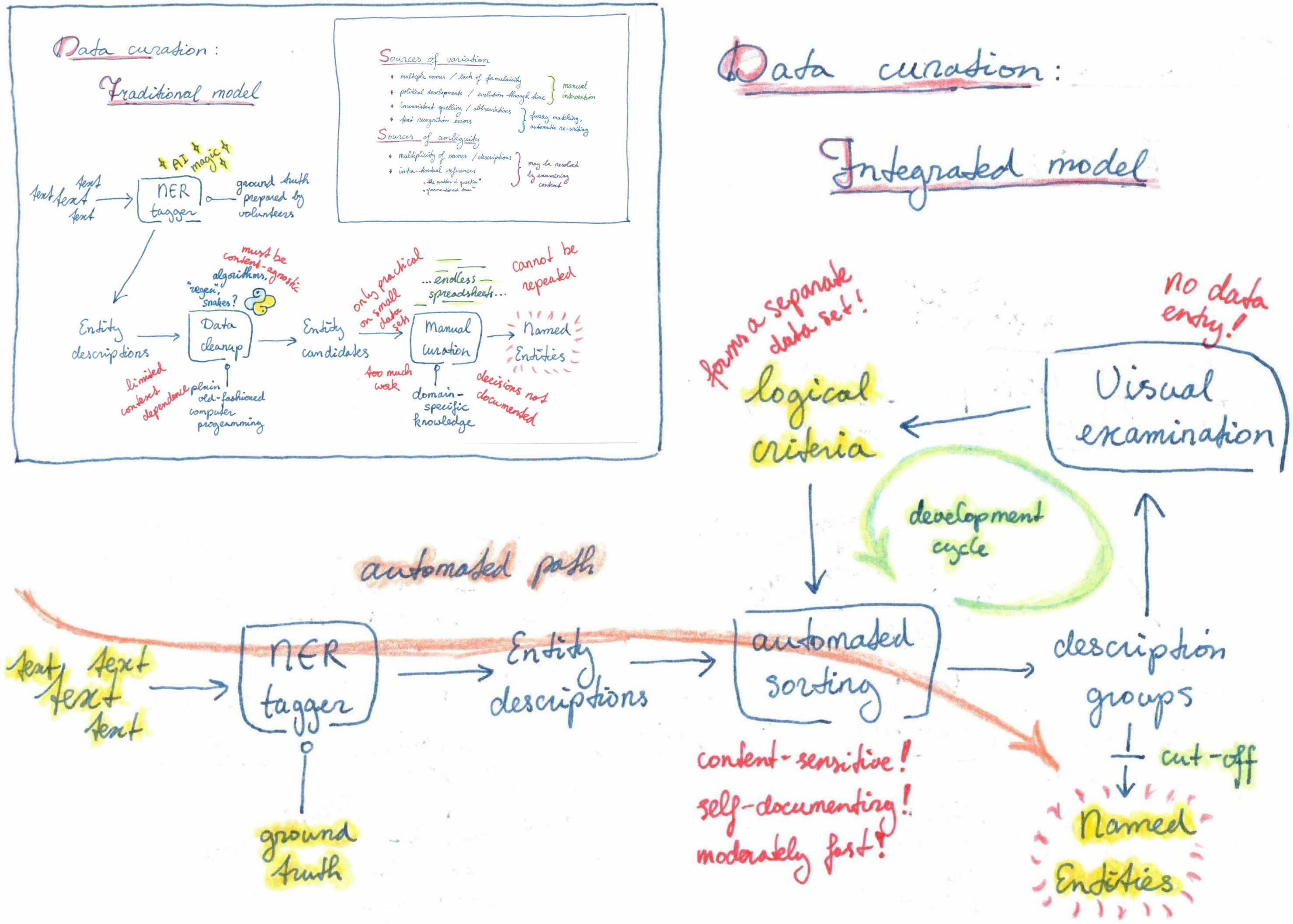


Data curation:

Integrated model






Data curation:
Traditional model
AI magic
text text text text
NER tagger
ground truth prepared by volunteers
Sources of variation
multiple names / lack of formulaicity
political developments / evolution through time
manual intervention
inconsistent spelling / abbreviations
text recognition errors
fuzzy matching, automatic re-writing
Sources of ambiguity
multiplicity of names / descriptions
intra-textual references
„the matter in question"
„aforementioned town"
may be resolved by examining context
must be content-agnostic
algorithms, "regex", snakes?
Entity descriptions
Data cleanup
Entity candidates
only practical on small data sets
...endless spreadsheets...
Manual curation
cannot be repeated
Named Entities
limited context dependence
plain old-fashioned computer programming
too much work
domain-specific knowledge
decisions not documented
Data curation:
Integrated model
forms a separate data set!
logical criteria
no data entry!
Visual examination
development cycle
automated path
text text text text
NER tagger
ground truth
Entity descriptions
automated sorting
content-sensitive!
self-documenting!
moderately fast!
description groups
cut-off
Named Entities

Example: Admiralities of the Republic

Once sorted in the box „Admirality",
divide again on the following keywords:

Dokkum

Harlingen

Vrieslandt

Rotterdam

Op de Maze

Amst[a-z]*

Example: Admiralities of the Republic

Once sorted in the bin „Admiralty",
divide again on the following keywords:

Dokkum
Harlingen
Vrieslandt
} Admiralty of Frisia

Rotterdam
Op de Maze
} Admiralty of Rotterdam

Amst[a-z]* — Admiralty of Amsterdam

Possible entities are known in advance.

Narrow semantic domain enables improvements over OCR/HTR

Example: sorting local governments

Borgermren ende Regierders der Stadt Amsterdam

Syndicus ende Raedt ~~der~~ van Geneve

hooch Baillu ende Schepenen van Gent

Regenten van Helmont, Quartiere van Peellandt,
Meyerye van 's Hertogenbosch

Example : sorting local governments

Borgermeen ende Regierders der Stadt Amsterdam

Syndicus ende Raedt ~~der~~ van Geneve

hooch Baillu ende Schepenen van Gent

Regenten van Helmont, Quartiere van Peellandt,
Meyerye van 's Hertogenbosch

Example: sorting local governments

functions in local governments

Borgermeen ende Regierders der Stadt Amsterdam → Amsterdam

Syndicus ende Raedt ~~der~~ van Geneve → Genève

hooch Baillu ende Schepenen van Gent → Gend

Regenten van Helmont, Quartiere van Peellandt,
Meyerye van 's Hertogenbosch → Helmond

Attributions → Organisations ← Locations

Conclusions

- Domain-specific NER tagging and training
  - Attributions are difficult
  - Ambiguity in instructions and in entities themselves
- Curation
  - Identification and resolution of entity references by successive grouping
  - (Logical) criteria for grouping form a separate dataset
  - Nesting of entity types and partial overlap can be powerful tools
- Analysis
  - Decomposing complex entities allows for combining multiple dimensions of analysis
- Project results will be published by December 2024!

Thank You!

We also thank the volunteers for their contributions to this project!

Questions?