The Treatment of Word Formation in the LiLa Knowledge Base

Eleonora Litta, **Marco Passarotti** and **Francesco Mambrini**

DeriMo 2019 | ÚFAL, Prague | 19-20 September 2019

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme – Grant Agreement No 769994

Research question
State of affairs

We have built and collected (for Latin and other languages):

- Textual Resources
- Lexical Resources
- NLP Tools

Research question
State of affairs

We have built and collected (for Latin and other languages):

- Textual Resources
- Lexical Resources
- NLP Tools

Scattered and unconnected

Research need

Making sense

To make sense of this quantity of empirical data:

Research need
Making sense

To make sense of this quantity of empirical data:

- to extract maximum benefit from our research investments

Research need

Making sense

To make sense of this quantity of empirical data:

- to extract maximum benefit from our research investments
- to impact and improve the life of Classicists through exploitable computational resources and tools

Research need
Making sense

To make sense of this quantity of empirical data:

- to extract maximum benefit from our research investments
- to impact and improve the life of Classicists through exploitable computational resources and tools

From Information to Knowledge

2018-2023

A collection of interoperable linguistics resources (and NLP tools) described with the same vocabulary for knowledge description

Interlinking as a Form of Interaction

LiLa is based on an ontology made of:

LiLa is based on an ontology made of:

- **Individuals**: instances of objects (one specific token, lemma etc.)

LiLa is based on an ontology made of:

- **Individuals**: instances of objects (one specific token, lemma etc.)
- **Classes**: types of objects/concepts (token, lemma, PoS etc.)

LiLa is based on an ontology made of:

- **Individuals**: instances of objects (one specific token, lemma etc.)
- **Classes**: types of objects/concepts (token, lemma, PoS etc.)
- **Data properties**: attributes that objects can/must have (morphological features for lemmas/tokens)

LiLa is based on an ontology made of:

- **Individuals**: instances of objects (one specific token, lemma etc.)
- **Classes**: types of objects/concepts (token, lemma, PoS etc.)
- **Data properties**: attributes that objects can/must have (morphological features for lemmas/tokens)
- **Object properties**: ways in which classes and individuals can be related to one another: RDF triples.
  Labels from a restricted vocabulary of knowledge description: `hasLemma`, `hasPoS`

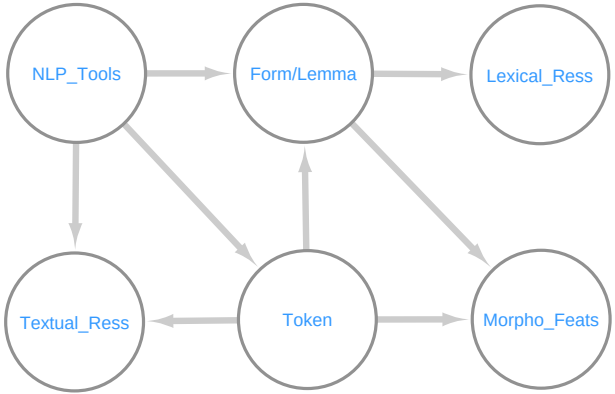
LiLa is based on an ontology made of:

- **Individuals**: instances of objects (one specific token, lemma etc.)
- **Classes**: types of objects/concepts (token, lemma, PoS etc.)
- **Data properties**: attributes that objects can/must have (morphological features for lemmas/tokens)
- **Object properties**: ways in which classes and individuals can be related to one another: RDF triples.
  Labels from a restricted vocabulary of knowledge description: `hasLemma`, `hasPoS`

Each component of the ontology is uniquely identified through a URI.

LiLa Knowledge Base

Lexically-based architecture and (meta)data sources



Word Formation Latin
recap

WFL: Word formation-based lexical resource for Classical Latin

Word Formation Latin
recap

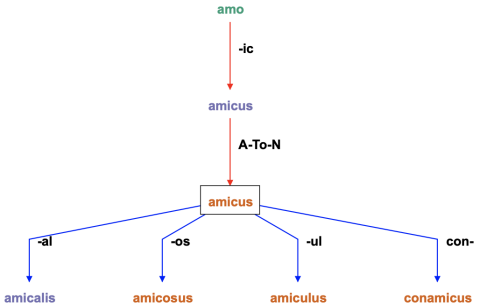
WFL: Word formation-based lexical resource for Classical Latin

- WFRs are modelled as **directed one-to-many input-output** relations between lemmas (based on I&A model of grammatical description)

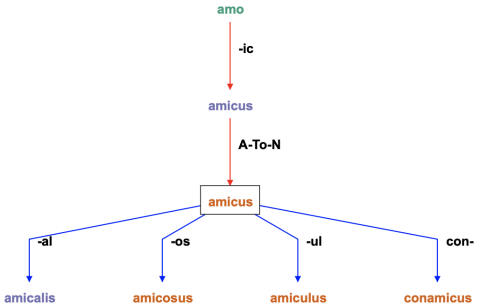
Word Formation Latin
recap

WFL: Word formation-based lexical resource for Classical Latin

- WFRs are modelled as **directed one-to-many input-output** relations between lemmas (based on I&A model of grammatical description)
- **Morphotactic** approach: each WF process is treated individually as the application of one single rule in a certain order



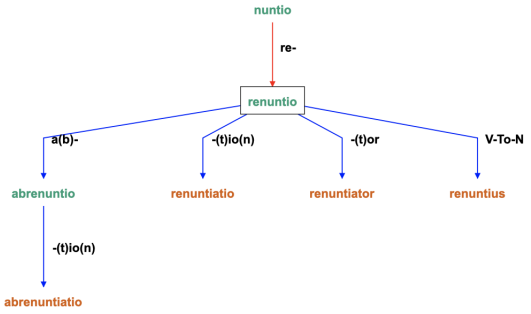
- Relationships between lemmas of the same “word formation family” are represented as the edges in a **directed graph** with a **hierarchical tree-like** structure



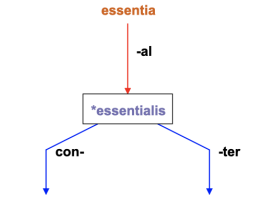
- Relationships between lemmas of the same “word formation family” are represented as the edges in a **directed graph** with a **hierarchical tree-like** structure
- A **node** is a lemma, and an **edge** is the WFR used to derive the output lemma from the input one, together with any affix

But: **directed graphs** are not completely satisfactory in representing the full range of relationships included within a word formation family.
Main problems:

- Directionality



- Non-linear derivations



Paradigmatic approach to WF: Requirements

Paradigmatic approach to WF: Requirements

- No directionality: necessary to accommodate those lemmas for which the derivational process is not of the simplex (or simpler) > complex type

Paradigmatic approach to WF: Requirements

- No directionality: necessary to accommodate those lemmas for which the derivational process is not of the simplex (or simpler) > complex type
- The CELL has a central role in the paradigm (predictability and regularity)

Paradigmatic approach to WF: Requirements

- No directionality: necessary to accommodate those lemmas for which the derivational process is not of the simplex (or simpler) > complex type
- The CELL has a central role in the paradigm (predictability and regularity)
- Each cell must be described in both its morphological characteristics and its semantic features, due to the underlying role of semantics in accounting for derivational processes

Word Formation in LiLa

Different approach to Word Formation:

Word Formation in LiLa

Different approach to Word Formation:

- Structure: **declarative** rather than procedural

Word Formation in LiLa

Different approach to Word Formation:

- Structure: **declarative** rather than procedural
- No directionality

Word Formation in LiLa

Different approach to Word Formation:

- Structure: **declarative** rather than procedural
- No directionality
- No morphotaxis.

Background Theory:
Construction Morphology (CxM)

- Construction: [co(n) [stell](a)(t)io]$_N$ (more specific)

Background Theory:
Construction Morphology (CxM)

- Construction: [co(n) [stell](a)(t)io]$_N$ (more specific)
- Schema [co(n)[x](t)io]$_N$ (more generalised)

Background Theory:
Construction Morphology (CxM)

- Construction: [co(n) [stell](a)(t)io]$_N$ (more specific)
- Schema [co(n)[x](t)io]$_N$ (more generalised)
- Constructions and schemas are word-based and declarative

Background Theory:
Construction Morphology (CxM)

- Construction: [co(n) [stell](a)(t)io]$_N$ (more specific)
- Schema [co(n)[x](t)io]$_N$ (more generalised)
- Constructions and schemas are word-based and declarative
- Perfect for LiLa => words are described in their formative elements, which can be organised into connected classes of objects into an ontology.

Word formation in LiLa

Three classes of objects:

Word formation in LiLa

Three classes of objects:

1. Lemmas

Word formation in LiLa

Three classes of objects:

1. Lemmas
2. Affixes (prefixes and suffixes)

Word formation in LiLa

Three classes of objects:

1. Lemmas
2. Affixes (prefixes and suffixes)
3. Bases (connectors between lemmas of the same WF family)

Word formation in LiLa

Three classes of objects:

1. Lemmas
2. Affixes (prefixes and suffixes)
3. Bases (connectors between lemmas of the same WF family)

Connected by three possibile relationships:

Word formation in LiLa

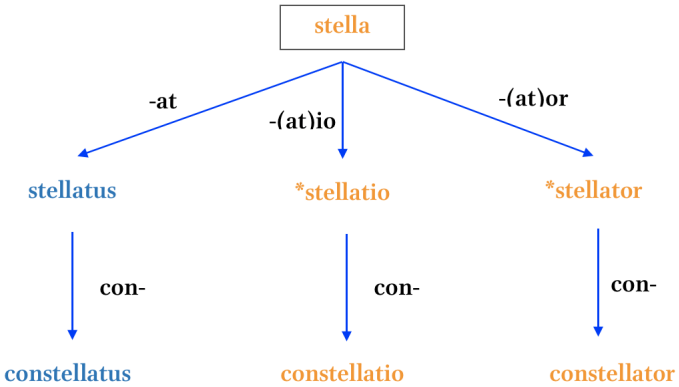
Three classes of objects:

1. Lemmas
2. Affixes (prefixes and suffixes)
3. Bases (connectors between lemmas of the same WF family)

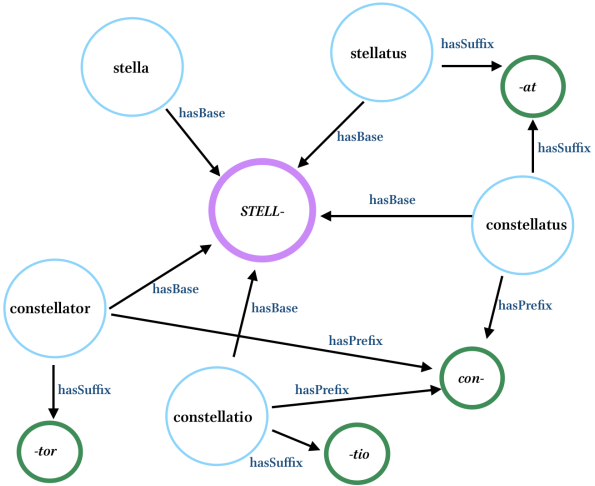
Connected by three possibile relationships:

1. hasPrefix
2. hasSuffix
3. hasBase

Stella - WFL


stella
-at
-(at)io
-(at)or
stellatus
*stellatio
*stellator
con-
con-
con-
constellatus
constellatio
constellator

STELL - LiLa


stella
stellatus
hasSuffix
-at
hasBase
hasBase
hasSuffix
STELL-
hasBase
constellatus
constellator
hasBase
hasBase
hasPrefix
hasPrefix
con-
hasSuffix
constellatio
hasPrefix
-tor
hasSuffix
-tio

LiLa triplestore available at:
https://lila-erc.eu/data/

Connecting WF info with various linguistic resources, e.g.

LiLa triplestore available at:
https://lila-erc.eu/data/

Connecting WF info with various linguistic resources, e.g.

- Find all occurrences of lemmas from the same WF family in the corpora connected in LiLa

LiLa triplestore available at:
https://lila-erc.eu/data/

Connecting WF info with various linguistic resources, e.g.

- Find all occurrences of lemmas from the same WF family in the corpora connected in LiLa
- Find all occurrences of nouns displaying agent/instrument and action suffixes (tio/tor) that govern verbs as subjects in the Latin treebanks connected in LiLa

LiLa triplestore available at:
https://lila-erc.eu/data/

Connecting WF info with various linguistic resources, e.g.

- Find all occurrences of lemmas from the same WF family in the corpora connected in LiLa
- Find all occurrences of nouns displaying agent/instrument and action suffixes (tio/tor) that govern verbs as subjects in the Latin treebanks connected in LiLa
- Count the frequency of the 15 most used affixes attached to nouns

Future Plans

Many!

- Find a way of defining and naming all "base" nodes
- Perhaps try to add word formation specific semantic information to the LiLa knowledge base
- Enlarge the lexical basis for which WF is provided with Medieval Latin lemmas contained in Lemlat.

Conclusions
Summing up

Added value of adding WFL to the LiLa Knowledge Base:

- allows for a better displayed, less assuming, less problematic way of describing words in their formative elements
- lets us connect a lexical resource with the realisation of its words inside texts.

Thank you!
Get in touch

The LiLa Team
Università Cattolica del Sacro Cuore
CIRCSE Research Centre

info@lila-erc.eu

https://github.com/CIRCSE

https://lila-erc.eu

@ERC_LiLa

Largo Gemelli 1, 20123 Milan, Italy

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme - Grant Agreement No. 769994.