Natural Language Interaction with Semantic Web Ontologies

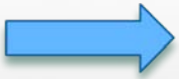
Gerasimos Lampouras

Department of Infomatics,
Athens Univesity of Economics and Business

Overall Motivation

- The **Semantic Web** is an effort to establish standards and mechanisms to allow computers to reason more easily about the semantics of the Web's resources (documents, data etc.).
- **Ontologies** play a central role in the **Semantic Web**.
  - They provide a **conceptualization** of a knowledge domain: **classes** of **individuals**, possible **relations** between them etc.
  - People unfamiliar with formal knowledge representation often have **difficulties understanding** ontology axioms (in OWL 2.0).

```
EquivalentClasses(:StEmilion
  ObjectIntersectionOf(:Bordeaux
    ObjectHasValue(:locatedIn :stEmilionRegion)
    ObjectHasValue(:hasColor :red)
    ObjectHasValue(:hasFlavor :strong)
    ObjectHasValue(:madeFrom
      :cabernetSauvignonGrape)
    ObjectMaxCardinality(1 :madeFrom)))
```



St. Emilion is a red strong Bordeaux. It is made from exactly one grape variety: Cabernet Sauvignon grapes.

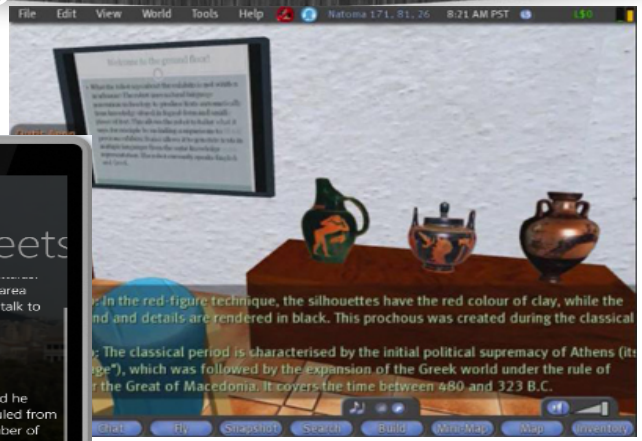
We consider methods that allow end-users to view **ontology-based knowledge representations** of the Semantic Web in the form of **automatically generated natural language texts**.

Usefulness

- Given an **OWL ontology** for a knowledge domain, one can **publish** on the Web **machine-readable statements** about the domain (e.g., available products, known diseases, their features or symptoms).
- **Natural Language Generation** can **produce texts** describing **classes** or **individuals** of the **ontology** (e.g., product descriptions, information about diseases) from the same statements.

- This way the **same information** becomes more easily accessible to both **computers** and **end-users**, which is one of the main goals of the Semantic Web.
  - Texts can be available in **different languages**.
  - **Changes** in the **OWL statements** can be **automatically reflected** in the texts by regenerating them.
  - Texts may be **adapted** to the **interests**, **knowledge background** of the **end-user**.
  - **Terminology** and **style** can be easily **maintained across texts** if needed.

Use Cases of Natural Language Generation



Outline

I. Natural Language Generation from ontologies

II. Generating texts with Integer Linear Programming

III. Extracting NLG resources from the Web

Natural Language Generation

- **Ontology verbalizers**: typically express descriptions in controlled, often **not entirely fluent** natural language primarily for the benefit of domain experts.
  - But they require **no resources** other than the ontology.

```
EquivalentClasses(:StEmilion
  ObjectIntersectionOf(:Bordeaux
    ObjectHasValue(:locatedIn :stEmilionRegion)
    ObjectHasValue(:hasColor :red)
    ObjectHasValue(:hasFlavor :strong)
    ObjectHasValue(:madeFrom
      :cabernetSauvignonGrape)
    ObjectMaxCardinality(1 :madeFrom)))
```

Verbalizer →

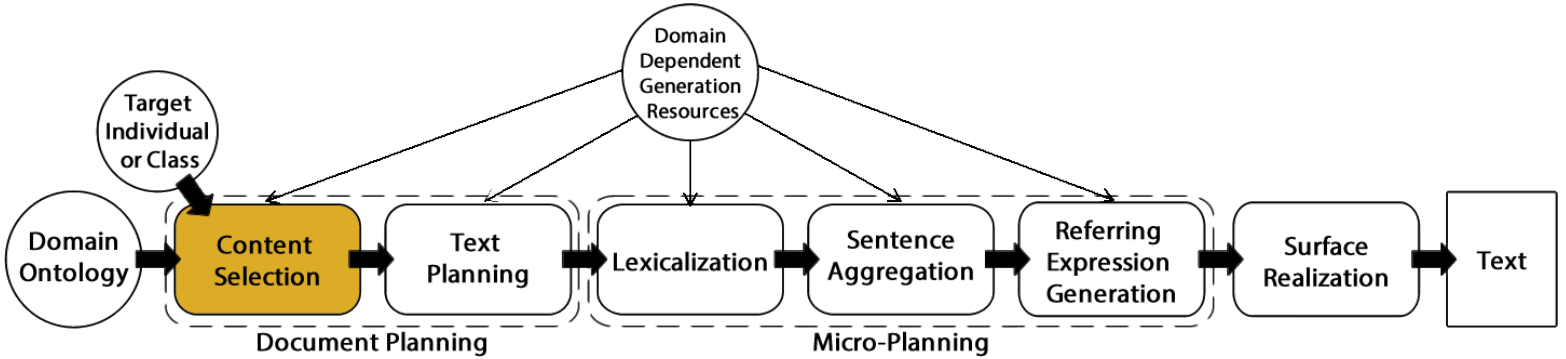
The *St Emilion* is a *Bordeaux*.
The *St Emilion* has as color *red*.
The *St Emilion* has as flavor *strong*.
…

NaturalOWL →

St. Emilion is a red strong Bordeaux.
…

- **NaturalOWL 2.0** is a **Natural Language Generation** system that aims to produce **fluent and coherent** multi-sentence texts describing classes or individuals of ontologies.
  - Improvement of the original **NaturalOWL**, previously developed at **AUEB.**
  - Supports both **English** and **Greek**.

Pipeline Architecture



<:StEmilion, isA, :Bordeaux>

<:StEmilion, :hasColor, :red>

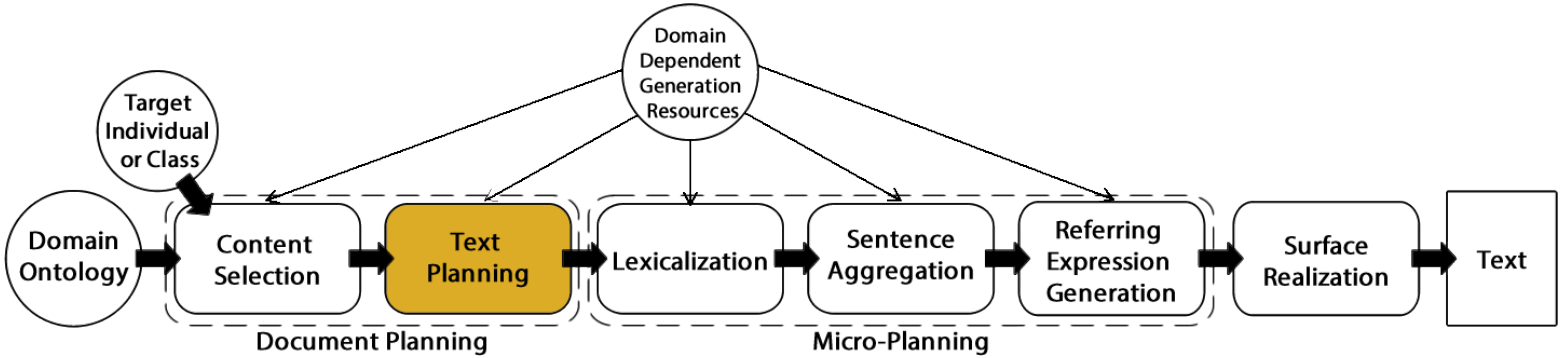
~~<:StEmilion, :locatedIn, :stEmilionRegion>~~

<:StEmilion, :madeFrom, :cabernetSauvignonGrape>

<:StEmilion, :hasFlavor, :strong>

<:StEmilion, maxCardinality(:madeFrom), :1>

Pipeline Architecture



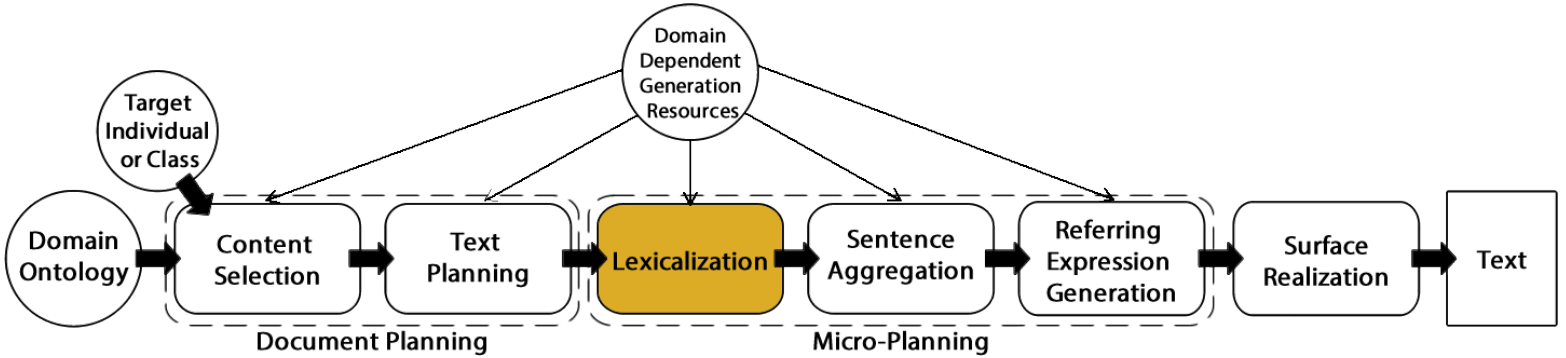
Description section

<:StEmilion, isA, :Bordeaux>
<:StEmilion, :hasColor, :red>
<:StEmilion, :hasFlavor, :strong>

Production section

<:StEmilion, :madeFrom, :cabernetSauvignonGrape>
<:StEmilion, maxCardinality(:madeFrom), :1>

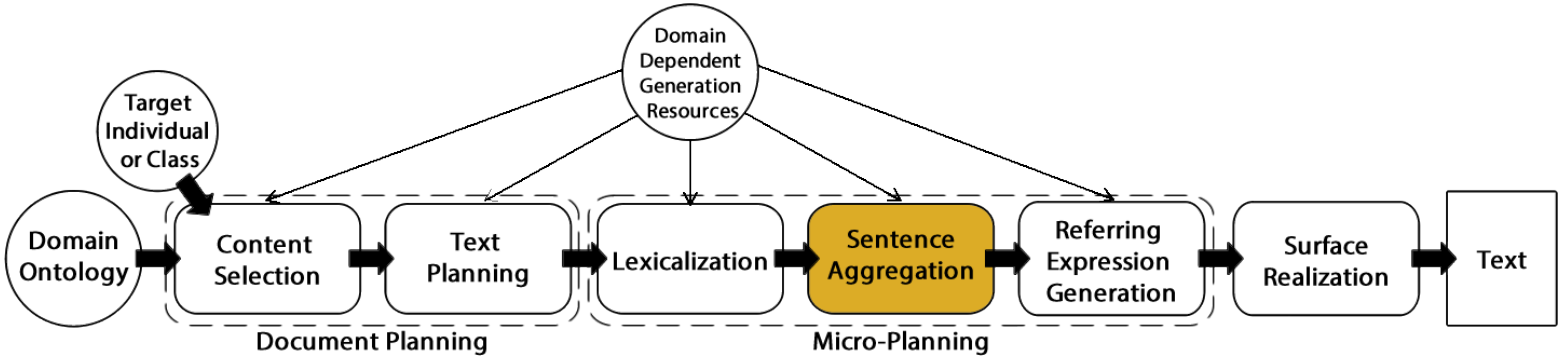
Pipeline Architecture



<:StEmilion, isA, :Bordeaux>
<:StEmilion, :hasColor, :red>
...
...

[:StEmilion] [toBe] [a kind of] [:Bordeaux]
[:StEmilion] [toBe] [:red]
...
...

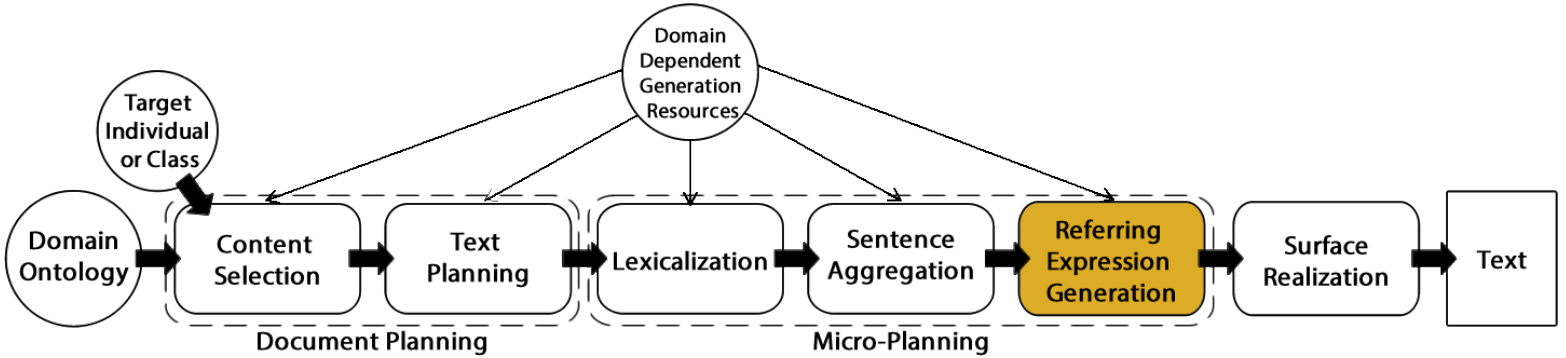
Pipeline Architecture



[:StEmilion] [toBe] [a kind of] [:Bordeaux]
[:StEmilion] [toBe] [:red]
...
...

[:StEmilion] [toBe] [a kind of] [:red] [:Bordeaux]
...
...

Pipeline Architecture



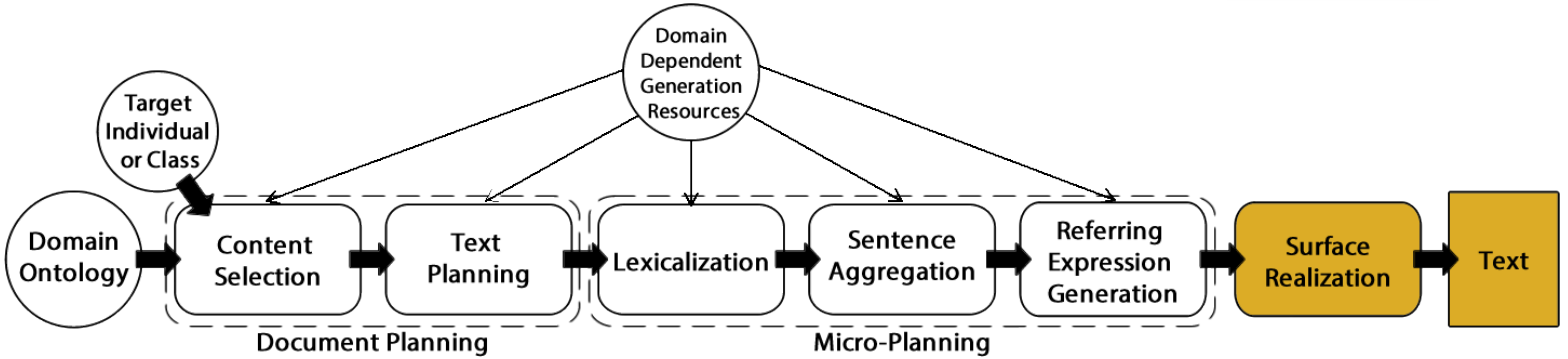
[:StEmilion] [toBe] [a kind of] [:red] [:Bordeaux]

...

[This] [toBe] [a kind of] [red] [Bordeaux]

...

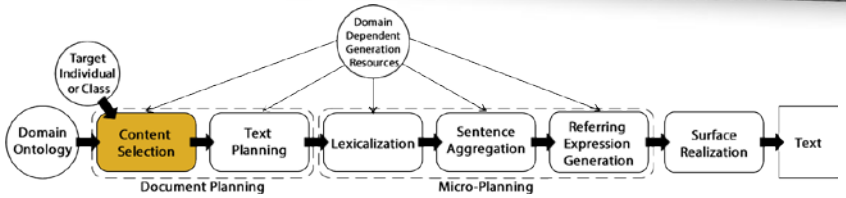
Pipeline Architecture



St. Emilion is a red strong Bordeaux.

It is made from exactly one grape variety: Cabernet Sauvignon grapes.

Content Selection



OWL Statements

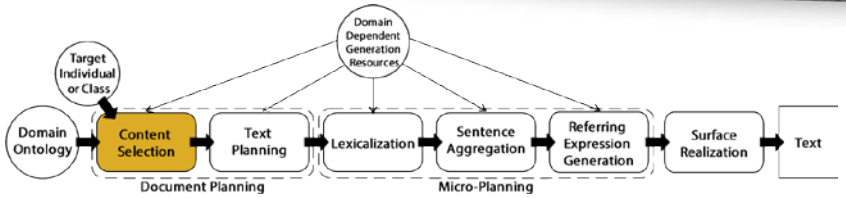
EquivalentClasses(:StEmilion
ObjectIntersectionOf(:Bordeaux
ObjectHasValue(:locatedIn :stEmilionRegion)
ObjectHasValue(:hasColor :red)
ObjectHasValue(:hasFlavor :strong)
ObjectHasValue(:madeFrom
:cabernetSauvignonGrape)
ObjectMaxCardinality(1 :madeFrom)))

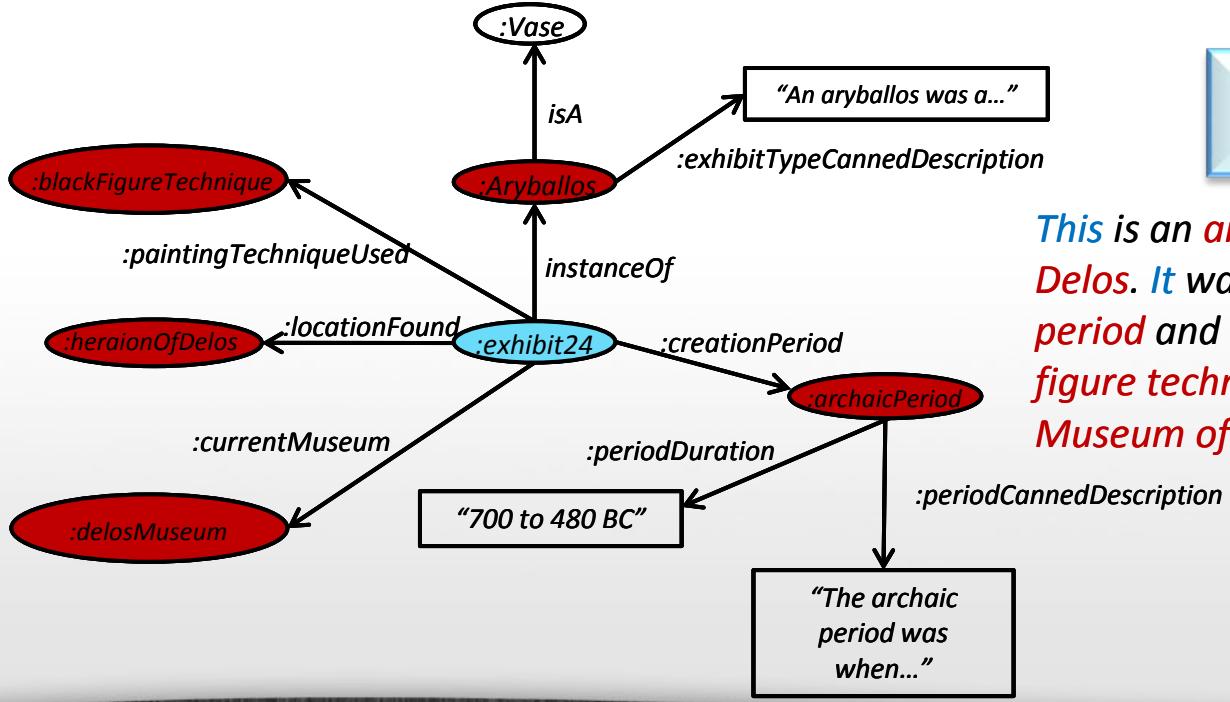
Message Triples <S, P, O>

<:StEmilion, isA, :Bordeaux>
<:StEmilion, :hasColor, :red>
<:StEmilion, :locatedIn, :stEmilionRegion>
<:StEmilion, :madeFrom, :cabernetSauvignonGrape>
<:StEmilion, :hasFlavor, :strong>
<:StEmilion, maxCardinality(:madeFrom), :1>

Message triples are not simply **RDF triples**, e.g. they may **contain property modifiers.** Every **message triple** corresponds to a sentence.

Content Selection

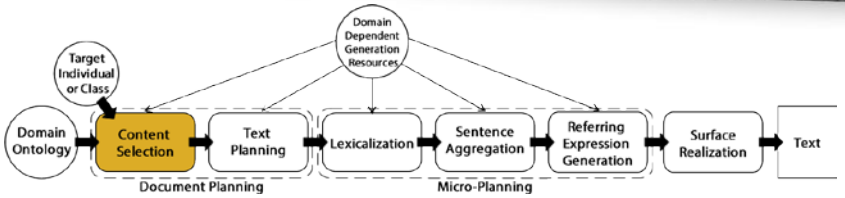


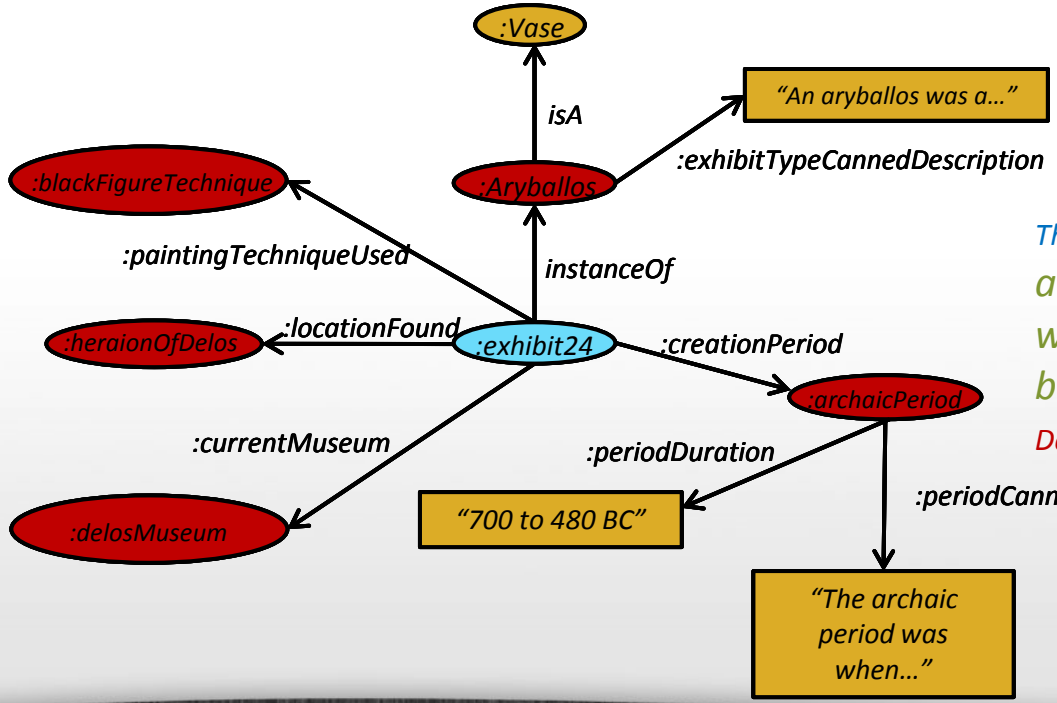


Text with only **first-level** targets.

This is an aryballos, found at the Heraion of Delos. It was created during the archaic period and it was decorated with the black-figure technique. It is currently in the Museum of Delos.

Content Selection

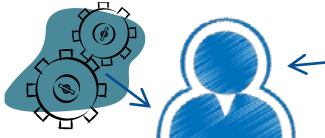




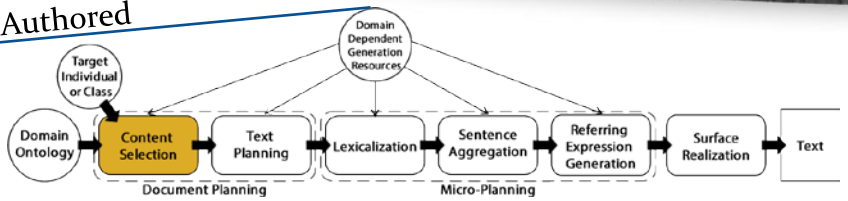
Text with up to **second-level** targets.

This is an aryballos, a kind of vase. An aryballos was a small spherical vase with a narrow neck, in which the athletes kept the oil they spread their bodies with. This aryballos was found at the Heraion of Delos and it was created during the archaic period. (...)

Content Selection – User Models



Authored



User Model

- Text preferences.
- Interest preferences.
- History of triples seen.

The **length** of the text in **triples**.

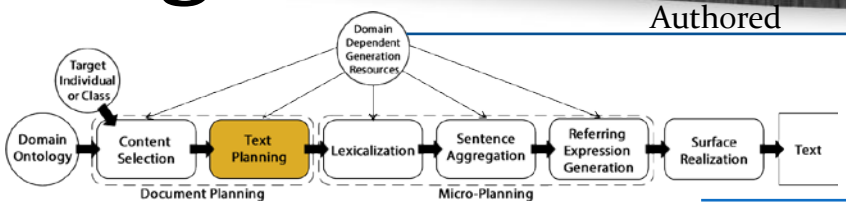
2 <:StEmilion, isA, :Bordeaux>
3 <:StEmilion, :hasColor, :red>
0 ~~<:StEmilion, :locatedIn, :stEmilionRegion>~~
2 <:StEmilion, :madeFrom, :cabernetSauvignonGrape>
2 ~~<:StEmilion, :hasFlavor, :strong>~~
1 <:StEmilion, maxCardinality(:madeFrom), :1>

<:StEmilion, isA, :Bordeaux>
<:StEmilion, :hasColor, :red>
<:StEmilion, :madeFrom, :cabernetSauvignonGrape>
<:StEmilion, maxCardinality(:madeFrom), :1>

Interest scores can be set on relation, class, and instance level:
1-3: levels of interest
0: do not report

Triples also have an **assimilation score**; how many times they should be repeated to the user before ignored.

Text Planning



<:StEmilion, isA, :Bordeaux>
<:StEmilion, :hasColor, :red>
<:StEmilion, :madeFrom, :cabernetSauvignonGrape>
<:StEmilion, :hasMaker, :Winery>
<:StEmilion, :hasFlavor, :strong>

Relation	Section	Order
isA	Description section	1
:hasColor	Description section	2
:hasFlavor	Description section	2
:madeFrom	Production section	2
:hasMaker	Production section	1

Global coherence happens through topical sections, which correspond to paragraphs.

Local coherence happens through ordering the triples within the sections.

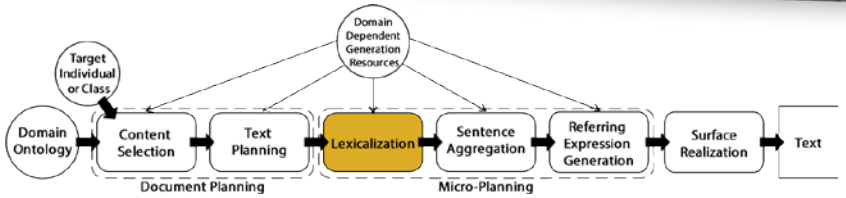
1 <:StEmilion, isA, :Bordeaux>
2 <:StEmilion, :hasColor, :red>
2 <:StEmilion, :hasFlavor, :strong>

1: Description section

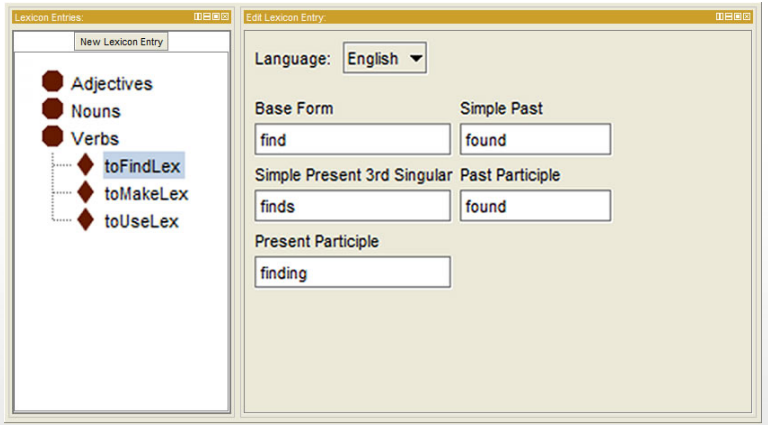
2 <:StEmilion, :madeFrom, :cabernetSauvignonGrape>
1 <:StEmilion, :hasMaker, :Winery>

2: Production section

Lexicalization – Lexicon



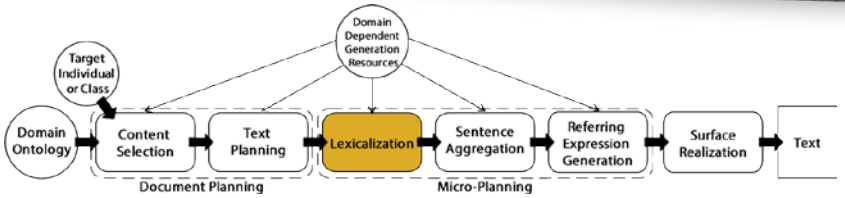
- During **lexicalization**, we turn the **output of content selection** to **abstract sentence specifications**.
- This stage requires the most **linguistic resources.**



- **Lexicon:** contains the **inflectional forms** of all the **adjectives**, **nouns** and **verbs** needed for generating the texts.
  - **Lexicon entries** are **multilingual** (English and Greek).

Lexicon entries can be created and edited through the **Protégé plugin** of **NaturalOWL**.

Lexicalization – Sentence Plans



- For every **relation**, one or more template-like **sentence plans** should be specified to indicate how **message triples** involving that relation can be **expressed in the text**.
- A **sentence plan** is a **sequence of slots**, along with instructions specifying how to fill them in.

Sentence plan for **:madeFrom** e.g. *<S, :madeFrom, O>*

$[ref(S)]^{1}_{nom}$ $[make]^{2}_{verb,\ passive,\ present,\ agr=1,\ polarity=+}$ $[from]^{3}_{prep}$ $[ref\ (O)]^{4}_{acc,\ plural}$

Referring expression to the **subject** of the message triple.

A **verb** from the **lexicon.**

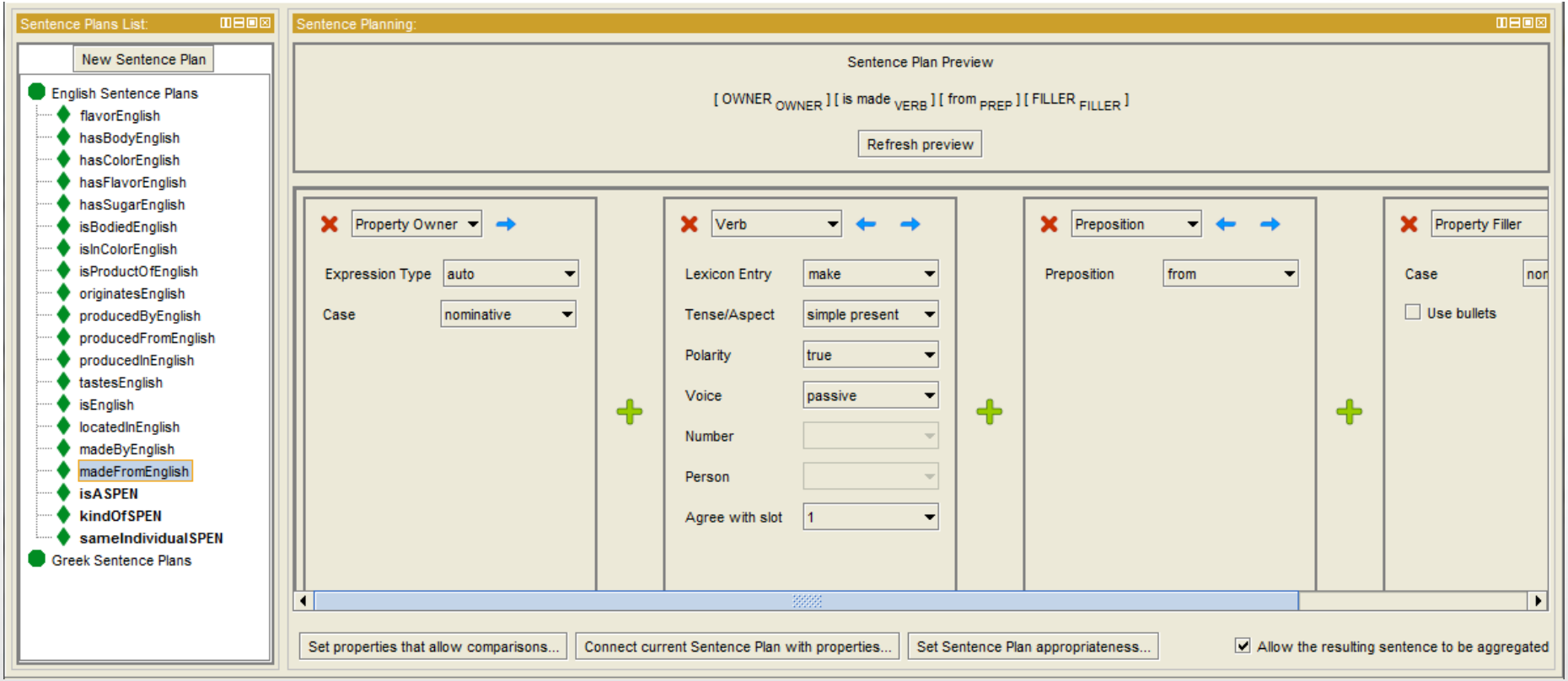
A **preposition.**

Referring expression to the **object** of the message triple. (or value if it is a data relation)

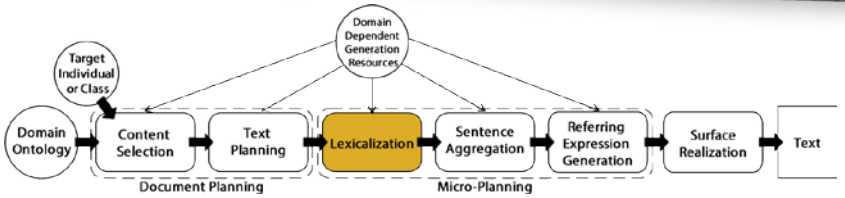
Wines are made from grapes.

Wines aren't made from grapes.

Lexicalization – Sentence Plans



Lexicalization – NL Names



- For every **individual or class**, one or more template-like **NL Names** should be specified to indicate how the **individual or class** can be **expressed in the text**.
- An **NL Name** is a **sequence of slots**, along with instructions specifying how to fill them in.

NL Names for *:aryballos* e.g. <*:aryballos, P, O*> and *:Iraion-Delos* e.g. <*S, P, :Iraion-Delos*>

$[]^{1}_{\text{article, indef, agr = 2}}$ $[\text{aryballos}]^{2}_{\text{headnoun, sing, nom}}$

$[]^{1}_{\text{article, def, agr = 2}}$ $[\textit{Heraion}]^{2}_{\text{headnoun, sing, nom}}$ $[\textit{of}]^{3}_{\text{prep}}$ $[\textit{Delos}]^{4}_{\text{noun, sing, nom}}$

Articles.

Nouns from the **lexicon.**

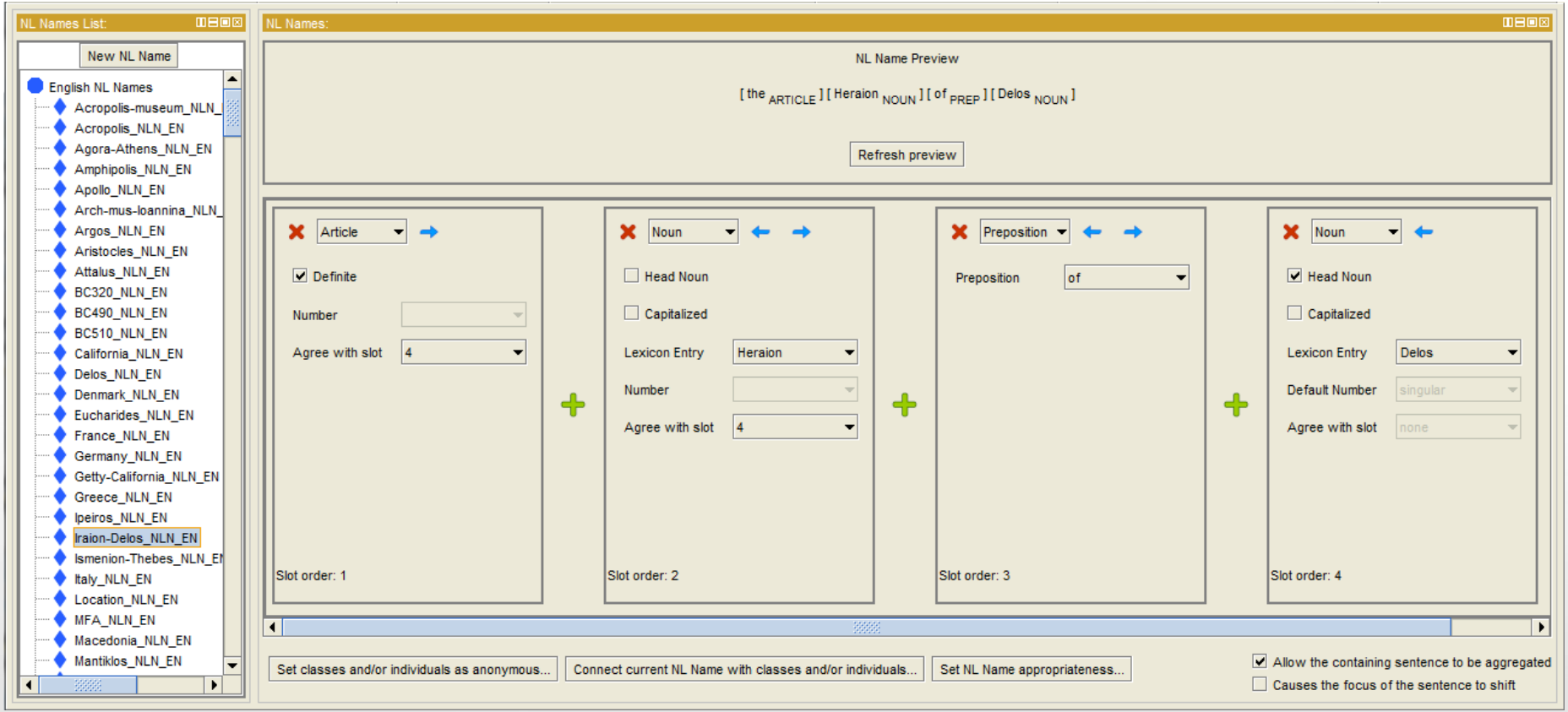
A **preposition.**

The **head's number** and **case** are taken to be the **number** and **case** of the overall **NLName**.

$[\textit{Cabernet}]_{\text{noun, agr=3}}$ $[\textit{Sauvignon}]_{\text{noun, agr=3}}$ $[\textit{grape}]_{\text{headnoun, sing, nom}}$

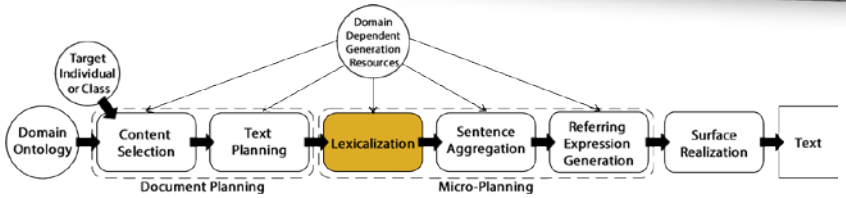
St. Emilion is made from Cabernet Sauvignon grapes.

Lexicalization – NL Names



Lexicalization

User Model



Appropriateness: some sentence plans/NL names may be more appropriate for adults vs. children.
History of sentence plans/NL names used.

<:StEmilion, :madeFrom, :cabernetSauvignonGrape> → [ref(S)] [make] [from] [ref (O)]

<:StEmilion, :hasColor, :red> → [ref(S)] [toBe] [ref (O)]

<:StEmilion, isA, :Bordeaux> → [ref(S)] [toBe] [a] [kind of] [ref (O)]

<:StEmilion, :hasFlavor, :strong> → [ref(S)][has flavor][ref (O)]

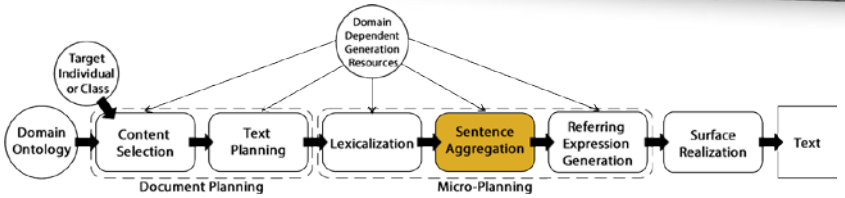
NaturalOWL has **built-in** sentence plans for **domain-independent relations** (e.g., isA, instanceOf).

The selection process is similar for **NL names.**

If no **authored sentence plans** is available **NaturalOWL** automatically generates one by **tokenizing** the **identifier of the relation**. This is similar to **verbalizers** and can only support **English** (not **Greek**).

Sentence Aggregation

User Model



The ordered triples are scanned greedily and a number of **aggregation rules** are applied to them.

[*ref(S)*] [*toBe*] [*a*] [*kind of*] [*ref (O)*] [*make*] [*from*] [*ref (O)*]

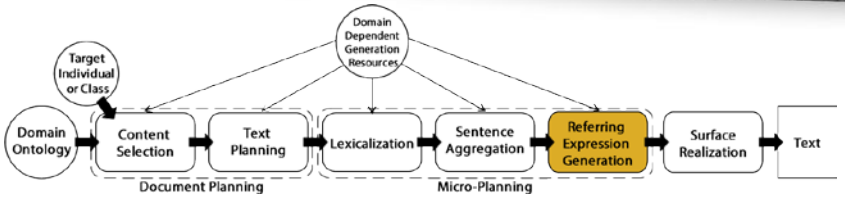
[*ref(S)*] [*toBe*] [*ref (O)*]

[*ref(S)*] [*toBe*] [*ref (O)*]

There is a **user limit** on **how many triples** can be aggregated into a **single sentence**.

In **sentence aggregation**, **triples** are combined to form **longer sentences**, in order to **avoid redundancy** and **increase the readability** of the final text.

Referring Expression Generation



In **referring expression generation**, the system generates each **referring expression slot** in each **sentence plan.**

To generate appropriate **pronouns** we need to know the gender of the **NL name**.

If the **user** does not want to use an **NL name** for a class or individual, it can be set as **anonymous**.

User Model

anonymous

This is an aryballos.
An aryballos is a kind of vase.
It was a small spherical vase with a narrow neck, in which the athletes kept the oil they spread their bodies with.
This aryballos was found at the Heraion of Delos.
It was created during the Archaic period.

NaturalOWL 2.0 vs. NaturalOWL

- Supports **OWL 2.0 ontologies**, as well as ontologies importing terms from other ontologies.
- Produces **descriptions of classes** as well as individuals.
- Supports **realization** of **complex axioms**, e.g. nested SubclassOf and ObjectIntersectionOf statements.

- **Reworked** and **extended** the **domain-dependent linguistic resources:**
  - Introduced **sections to text planning**.
  - Introduced **adjectives and verbs to the lexicon**.
  - Extended **sentence plan slots** and **options (tenses, polarity, etc.)** and introduced **agreement** functionality.
  - Introduced **NL names** and **anonymity** functionality.
  - Introduced more **aggregation** and **user modeling** options.
- The **linguistic resources** are represented as **OWL 2.0 ontologies,** making it easier to **share** and **reuse** them with other ontologies.
- The **linguistic resources** (tokenized identifiers) can be **automatically generated** if **no authored** resources are available.

- New **authoring tool**, supporting all linguistic resources, available as a **Protégé 4 plugin**.
- **Experiments** that examined the **quality** of the **generated descriptions**.

Trials – Wine Ontology

- **Wine Ontology:** Contains axioms and expressions covering almost all the functionality of OWL 2.
  - We generated texts that describe **43 classes** and **52 individuals**.

Generation Resources	
Sentence plans	*5*
Lexicon entries	*67*
Natural language names	*41*
Sections	*2*
Relation assignments to sections	*7*
Interest score assignments	*8*

- We **experimented** with the following **systems**:
  - **OWL verbalizer** of the **SWAT project**[1].

    The Foxen Chenin Blanc is a chenin blanc. The Foxen Chenin Blanc has as body Full. The Foxen Chenin Blanc has as flavor Moderate. The Foxen Chenin Blanc has as maker Foxen. The Foxen Chenin Blanc has as sugar Dry. The Foxen Chenin Blanc is located in the Santa Barbara Region.
  - **NaturalOWL (-)** system.

    Foxen Chenin Blanc is a Chenin Blanc. Foxen Chenin Blanc has sugar dry, Foxen Chenin Blanc has body full and Foxen Chenin Blanc has flavor moderate. Foxen Chenin Blanc has maker Foxen and Foxen Chenin Blanc located in the Santa Barbara County.
  - **NaturalOWL (+)** system.

    This wine is a moderate, dry Chenin Blanc. It has a full body. It is made by Foxen in the Santa Barbara County.

[1]http://swat.open.ac.uk/tools/

Trials – Wine Ontology Results

- We showed the generated **texts** of the three systems to **10 human judges**.
  - In a pilot study, we calculated strong **inter-annotator agreement** between 2 judges (**sample Pearson correlation r ≥ 0,91**) on a sample of **30 texts** (10 from each system).
- Judges were asked to **score** each of the following **criteria** from **1: disagreement** to **3: agreement**.

Criteria	SWAT	NaturalOWL (-)	NaturalOWL (+)
Sentence fluency	2.00 ± 0.15	1.76 ± 0.15	**2.80 ± 0.10**
Referring expressions	1.40 ± 0.13	1.15 ± 0.09	**2.72 ± 0.13**
Text structure	2.15 ± 0.16	2.20 ± 0.16	**2.94 ± 0.05**
Clarity	2.66 ± 0.13	2.55 ± 0.13	**2.74** ± 0.11
Interest	2.30 ± 0.15	2.14 ± 0.16	**2.68 ± 0.12**

The difference between systems whose 95% confidence intervals are bold has been shown to be statistically significant (t-test, a = 0.05).

Trials – Consumer Electronics Ontology

- **Consumer Electronics Ontology**
  - We generated texts for
    - 30 **development** individuals (10 cameras, 10 camcorders, 10 printers)
    - 30 **test** individuals (10 cameras, 10 camcorders, 10 printers)

Generation Resources	English	Greek
Sentence plans	42	42
Lexicon entries	19	19
Natural language names	36	36
Sections	6	
Relation assignments to sections	42	
Interest score assignments	12	

- The **domain-dependent linguistic resources** were authored by considering the descriptions of the **development individuals** exclusively.
  - **No linguistic resources** were authored especially for the **test individuals**.
  - We wanted to simulate a scenario where **new individuals** may be added to an ontology, but **no further linguistic resources** are authored.

- **Domain-dependent linguistic resources** were authored for both **English** and **Greek**.

Trials – Consumer Electronics Ontology Results

- We showed the generated **English texts** of the three systems to **10 human judges**.
- Judges were asked to **score** from **1: disagreement** to **3: agreement**.

Development Texts

Criteria	SWAT	NaturalOWL (-)	NaturalOWL (+)
Sentence fluency	1.97 ± 0.15	1.93 ± 0.27	**2.90 ± 0.08**
Referring expressions	1.10 ± 0.06	1.10 ± 0.11	**2.87 ± 0.08**
Text structure	1.67 ± 0.15	1.33 ± 0.19	**2.97 ± 0.04**
Clarity	1.97 ± 0.15	2.07 ± 0.26	**3.00 ± 0.00**
Interest	1.77 ± 0.14	1.73 ± 0.29	**3.00 ± 0.00**

The difference between systems whose 95% confidence intervals are bold has been shown to be statistically significant (t-test, a = 0.05).

Trials – Consumer Electronics Ontology Results

- We showed the generated **English texts** of the three systems to **10 human judges**.
- Judges were asked to **score** from **1: disagreement** to **3: agreement**.

Test Texts

Criteria	SWAT	NaturalOWL (-)	NaturalOWL (+)
Sentence fluency	2.03 ± 0.15	1.87 ± 0.15	**2.87 ± 0.08**
Referring expressions	1.10 ± 0.06	1.10 ± 0.06	**2.87 ± 0.08**
Text structure	1.57 ± 0.13	1.37 ± 0.12	**2.93 ± 0.05**
Clarity	2.07 ± 0.15	1.93 ± 0.15	**2.97 ± 0.04**
Interest	1.83 ± 0.17	1.60 ± 0.14	**2.97 ± 0.04**

The difference between systems whose 95% confidence intervals are bold has been shown to be statistically significant (t-test, a = 0.05).

Trials – Consumer Electronics Ontology Results

- We showed the generated **Greek texts** of the three systems to **10 human judges**.
- Judges were asked to **score** from **1: disagreement** to **3: agreement**.

Greek Texts

Criteria	NaturalOWL (+) development data	NaturalOWL (+) test data
Sentence fluency	**2.87 ± 0.12**	**2.83 ± 0.09**
Referring expressions	**2.77 ± 0.20**	**2.80 ± 0.11**
Text structure	**3.00 ± 0.00**	**3.00 ± 0.00**
Clarity	**3.00 ± 0.00**	**2.93 ± 0.05**
Interest	**2.97 ± 0.06**	**3.00 ± 0.00**

The difference between systems whose 95% confidence intervals are bold has been shown to be statistically significant (t-test, a = 0.05).

Ablation Trials

- In these trials, we studied how the **quality of the generated texts** is affected when **various components** and **linguistic resources** of **NaturalOWL** are gradually **removed**.
- We used **45 new test** products from the Consumer Electronics Ontology
  - Judges were asked to **score** from **1: strong disagreement** to **5: strong agreement**.

System Configuration	Sentence Fluency	Ref. Expressions	Text Structure	Clarity	Interest
NaturalOWL (+)	**4.80 ± 0.12**	**5.00 ± 0.00**	**4.82 ± 0.15**	**4.78 ± 0.12**	**4.89 ± 0.09**
- interest scores	4.53 ± 0.16	4.95 ± 0.06	4.78 ± 0.12	4.62 ± 0.17	4.20 ± 0.19
- ref. expr. gen.	3.93 ± 0.28	**1.53 ± 0.22**	4.80 ± 0.12	4.51 ± 0.24	4.07 ± 0.22
- NL names	3.71 ± 0.29	1.48 ± 0.21	4.71 ± 0.15	4.24 ± 0.25	3.98 ± 0.26
- aggregation	3.64 ± 0.33	1.33 ± 0.19	4.67 ± 0.16	4.24 ± 0.25	3.93 ± 0.26
- sentence plans	**2.07 ± 0.37**	1.33 ± 0.19	4.60 ± 0.18	**2.49 ± 0.36**	**2.38 ± 0.35**
- sections, ordering	1.89 ± 0.36	1.33 ± 0.19	**1.53 ± 0.24**	2.33 ± 0.33	1.89 ± 0.28

Summary and Contribution of this Section

- We presented **NaturalOWL 2.0,** an open-source **Natural Language Generation** system that produces **fluent and coherent** multi-sentence **English** and **Greek** texts describing classes and individuals of **OWL ontologies**.
  - It is the first complete, general-purpose **NLG system for OWL ontologies**.
  - The **domain-dependent linguistic resources** are also modeled as **OWL ontologies**.

- **Experiments** with **the Wine, Consumer Electronics and Disease ontologies** showed that:
  - **NaturalOWL**, compared to simpler **OWL verbalizers**, can produce **significantly better** natural language descriptions of classes and individuals, provided that appropriate domain-dependent generation resources are available,
  - the descriptions can be **generated in more than one languages**, again provided that appropriate resources are available,
  - but authoring **domain-dependent linguistic resources** can be **tedious** and **time consuming**.

- **Ablation experiments** examined the contribution of the **various components** of the **generation pipeline** and the **domain-dependent linguistic resources** to the **quality** of the generated texts. **Sentence plans** and **NL names** seem to have the greatest effect, followed by the **ordering resources**.

Summary and Contribution of this Section (2)

Publications

I. Androutsopoulos, G. Lampouras and D. Galanis,
"Generating Natural Language Descriptions from OWL Ontologies: the NaturalOWL System".
Journal of Artificial Intelligence Research, 48:671-715, 2013.

I. Androutsopoulos, G. Lampouras and D. Galanis,
"Generating Natural Language Descriptions from OWL Ontologies: A Detailed Presentation of the NaturalOWL System".
Technical Report, Natural Language Processing Group, Department of Informatics, Athens University of Economics and Business, 2012.

D. Galanis, G. Karakatsiotis, G. Lampouras and I. Androutsopoulos,
"An Open-Source Natural Language Generator for OWL Ontologies and its Use in Protege and Second Life".
System demonstration, *12th Conference of the European Chapter* of the *Association for Computational Linguistics* (EACL 2009), Athens, Greece, 2009.

S. Konstantopoulos, A. Tegos, D. Bilidas, I. Androutsopoulos, G. Lampouras, P. Malakasiotis, C. Matheson and O. Deroo,
"Adaptive Natural Language Interaction".
System demonstration, *12th Conference of the European Chapter* of the *Association for Computational Linguistics* (EACL 2009), Athens, Greece, 2009.

G. Karakatsiotis, D. Galanis, G. Lampouras and I. Androutsopoulos,
"NaturalOWL: Generating Texts from OWL Ontologies in Protege and in Second Life".
System demonstration, *18th European Conference on Artificial Intelligence* (ECAI 2009), Patras, Greece, 2008.

Outline

I. Natural Language Generation from ontologies

II. Generating texts with Integer Linear Programming

III. Extracting NLG resources from the Web

Motivation

- **Natural Language Generation** systems usually adopt **pipeline architectures**, which may **not lead** to **optimal texts,** in terms of quality or length.
  - **Space** can be **limited** or **expensive** (e.g. smartphones, advertisements).

<:StEmilion :isA :Bordeaux>
<:StEmilion :hasColor :red>
<:StEmilion :hasFlavor :strong>

[:StEmilion] [toBe] [a kind of] [:Bordeaux]
[:StEmilion] [toHave] [:red] [color]
[:StEmilion] [toHave] [:strong] [flavor]

St. Emilion is a kind of Bordeaux.
It has red color and strong flavor.

triples/words = 3/14 ≅ **0.21**

[:StEmilion] [toBe] [a kind of] [:Bordeaux]
[:StEmilion] [toBe] [:red]
[:StEmilion] [toBe] [:strong]

St. Emilion is a kind of red, strong Bordeaux.

triples/words = 3/9 ≅ **0.33**

Motivation

- **Natural Language Generation** systems usually adopt **pipeline architectures**, which may **not lead** to **optimal texts,** in terms of quality or length.
  - **Space** can be **limited** or **expensive** (e.g. smartphones, advertisements).

<:wine :isA :Riesling>
<:wine :hasBody :full>
<:wine :hasFlavor :moderate>
<:wine :hasSugar :sweet>
<:wine :hasMaker :Corbans>

<:wine :isA :Riesling>
<:wine :hasBody :full>
<:wine :hasFlavor :moderate>
<:wine :hasMaker :Corbans>

This is a full Riesling and it has moderate flavor. It is produced by Corbans.

triples/words = 4/15 ≅ **0.27**

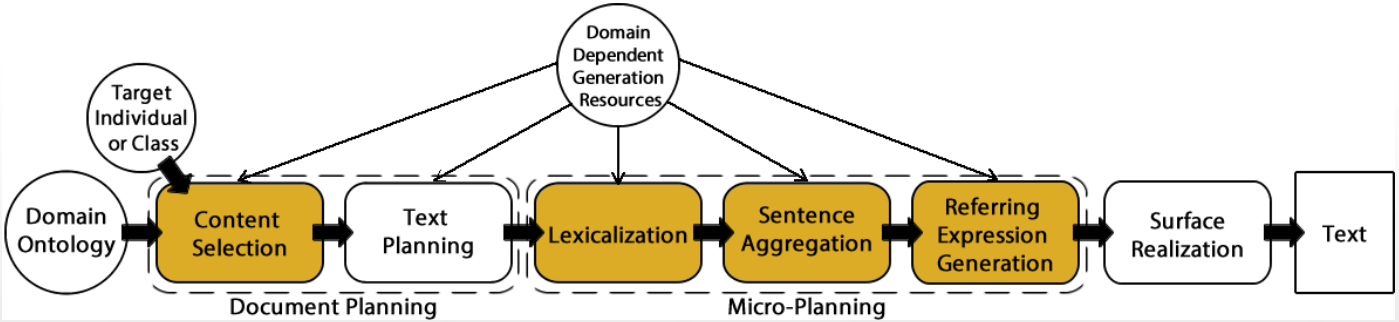
<:wine :isA :Riesling>
<:wine :hasBody :full>
<:wine :hasFlavor :moderate>
<:wine :hasSugar :sweet>

This is a full sweet moderate Riesling.

triples/words = 4/7 ≅ **0.57**

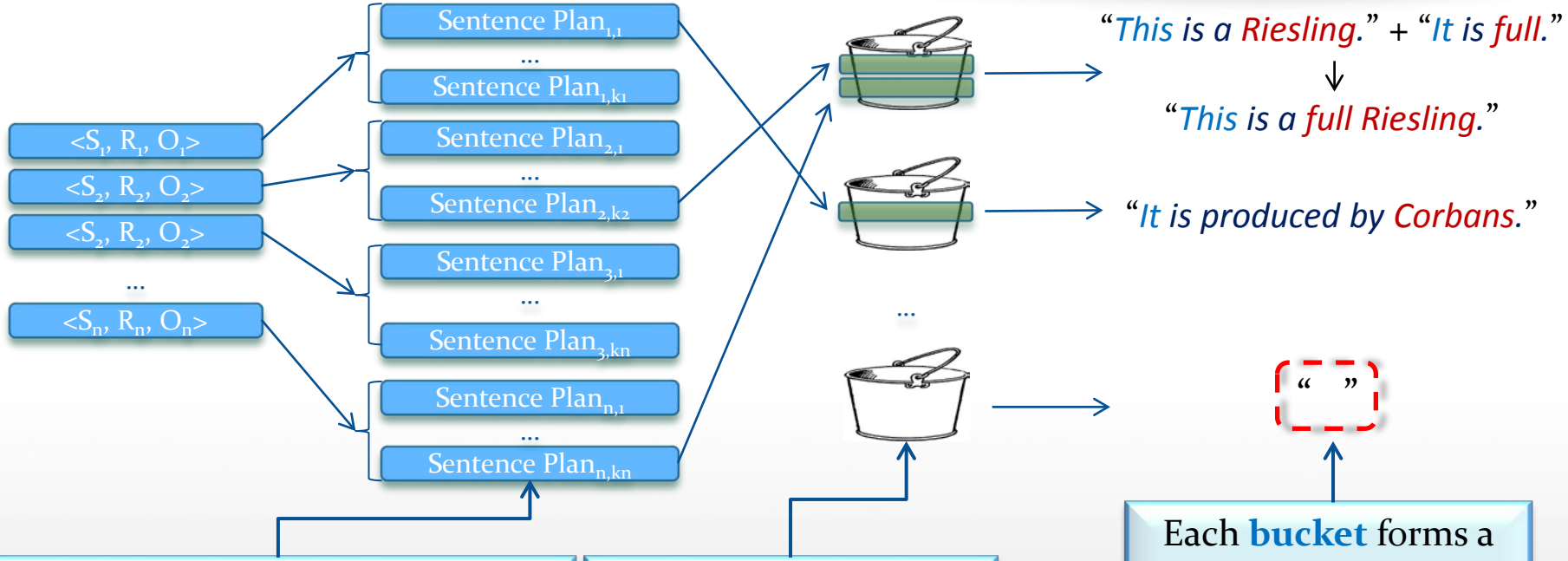
Global Optimization Approach

- Our **global optimization** approach combines:
  - **content selection**,
  - **lexicalization**,
  - and **aggregation**.
- An **extended version** also covers:
  - a limited form of **referring expression generation**.



- We use **Integer Linear Programming** (**ILP**).
  - **Binary LP**: all the **variables** are **binary (0/1)**.

Formulation



Each triple can be realized through different **sentence plans.**

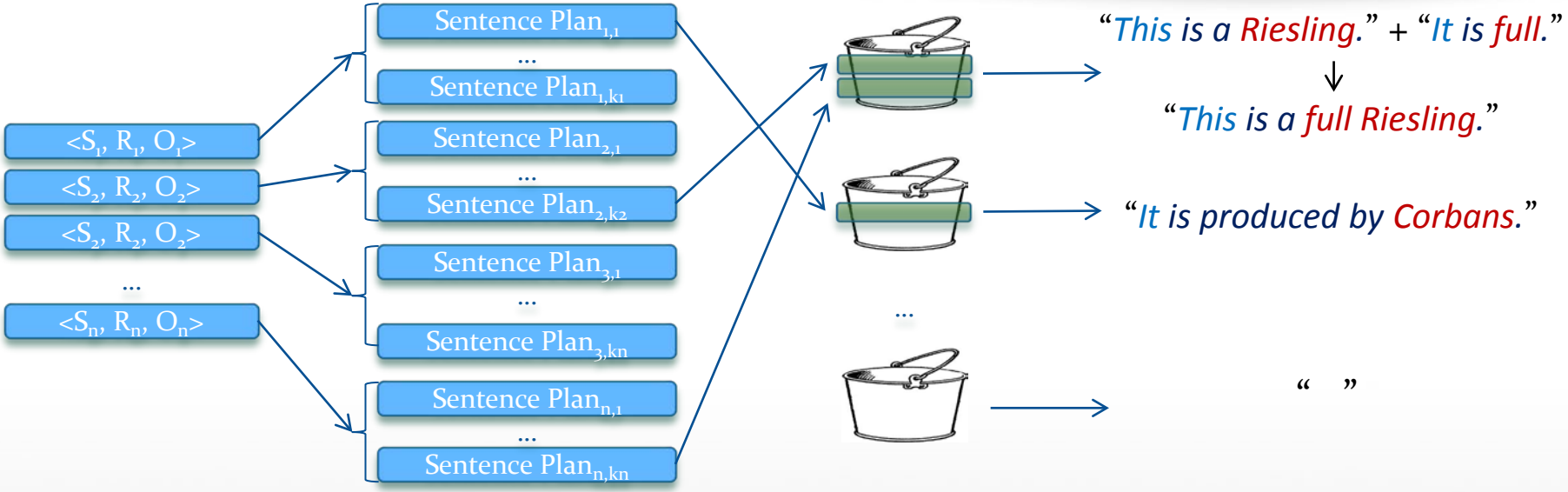
Each bucket may contain **o to n triples.**

Each **bucket** forms a single **sentence** in the final text.

Elements: every distinct filled-in slot in the sentence plans.

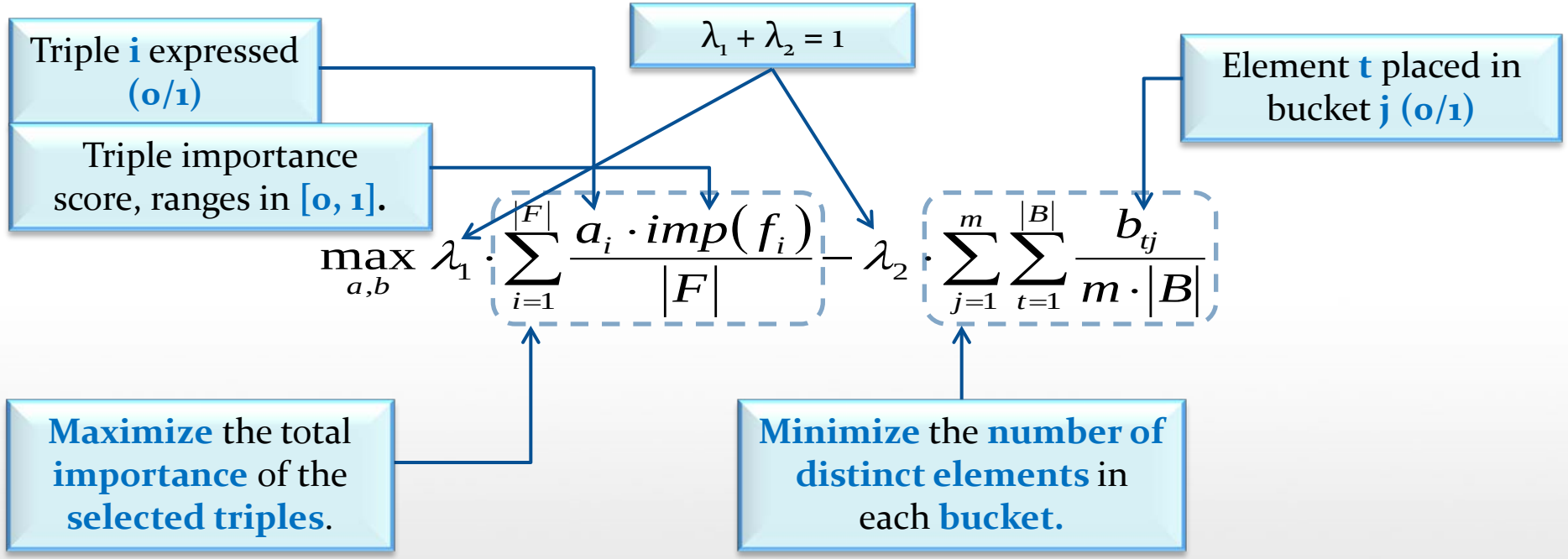
Empty buckets form **no** sentence.

Formulation



- We want to **maximize** the total **importance** of the **selected triples**.
  - Or the **number** of selected triples, if all triples are **equally important**.
- We want to **minimize** the number of **distinct elements** in each **bucket**.
  - To minimize the **approximate length** of the corresponding sentence.
  - And favor buckets that **aggregate well**.

ILP Objective function



ILP Constraints

Subject to:

$$a_i = \sum_{j=1}^{m} \sum_{k=1}^{|P_i|} l_{ikj}, \text{ for } i = 1, \ldots, n$$

Each **selected triple** must be expressed by **exactly one sentence plan** and placed in **exactly one bucket.**

$$\sum_{e_t \in B_{ik}} b_{tj} \geq |B_{ik}| \cdot l_{ikj}, \text{ for } \begin{matrix} i = 1, \ldots, n \\ j = 1, \ldots, m \\ k = 1, \ldots, |P_i| \end{matrix}$$

Constrains to ensure **consistency** between **sentence plans** and **elements**.

$$\sum_{p_{tk} \in P_{(e_t)}} l_{ikj} \geq b_{tj}, \text{ for } \begin{matrix} t = 1, \ldots, |B| \\ j = 1, \ldots, m \end{matrix}$$

ILP Constraints (2)

$$\sum_{t=1}^{|B|} b_{tj} \leq B_{max}, \text{ for } j = 1, \ldots, m$$

Each **bucket** cannot contain more than B_{max} **elements**.

In effect this limits the **maximum length** of an **aggregated sentence**.

$$\sum_{k=1}^{|P_i|} l_{ikj} + \sum_{k'=1}^{|P_{i'}|} l_{i'k'j} \leq 1, \text{ for } \begin{array}{r} j = 1, \ldots, m, i = 2, \ldots, n \\ i' = 1, \ldots, n-1; i \neq i' \\ \text{section}(f_i) \neq \text{section}(f_{i'}) \end{array}$$

Triples from **different sections cannot** be placed in the **same bucket**.

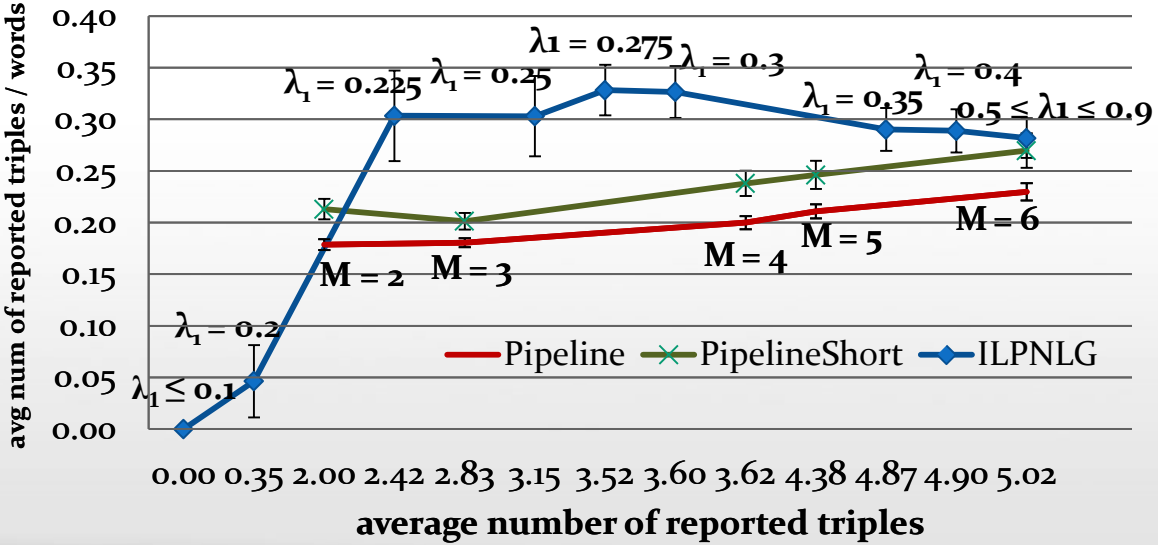
Evaluation Setup

- **Wine Ontology**
  - **52 individuals** with an **average of 5 triples** (max of 6 triples) per individual.
  - The generation resources include **3 sentence plans per relation**.
  - The **importance** scores of all triples were **set to 1**.

- We **experimented** with the following **systems**:
  - **PIPELINE** system.
    - Original **NaturalOWL** system using the **pipeline** architecture.
    - Selects **randomly** among the **available sentence plans**.
  - **PIPELINESHORT** system.
    - Same as **PIPELINE**, but always selects the **shortest (in elements)** sentence plan available.
  - **ILPNLG** system.
    - We modified **content selection**, **lexicalization**, and **aggregation** of the **NaturalOWL** system to use our **ILP model**.
    - We use **branch-and-cut** (GLPK) to solve the ILP model.

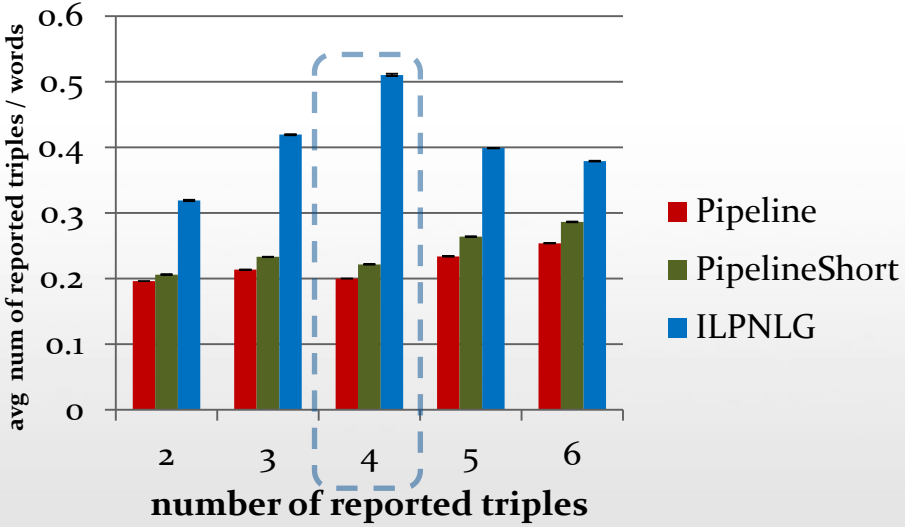
Wine Ontology Results

- **PIPELINE:**
  - **PIPELINE** has a parameter **M** specifying the **target number of triples** expressed by each text. We set **M = 2,3,4,5,6**.
- **ILPNLG:**
  - We generated texts for different values **λ1 (λ2 = 1 – λ1)** resulting in different numbers of reported triples.



Wine Ontology Results

- **PIPELINE:**
  - **PIPELINE** has a parameter **M** specifying the **target number of triples** expressed by each text. We set **M = 2,3,4,5,6**.
- **ILPNLG:**
  - We generated texts for different values **λ_1 ($\lambda_2 = 1 - \lambda_1$)** resulting in different numbers of reported triples.



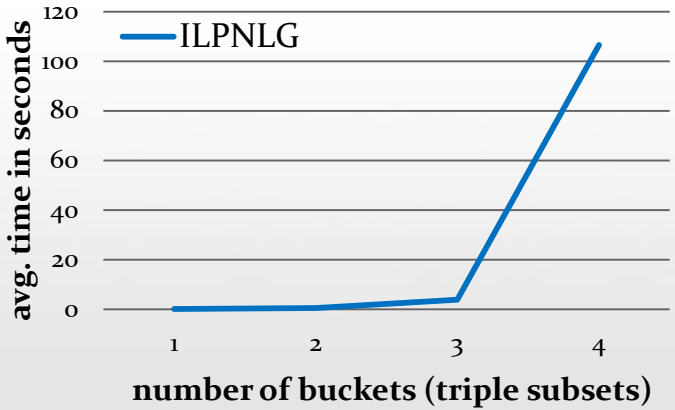
Each **set** compares all **texts** of the **same triple size** across different values of **M** and **λ_1**.

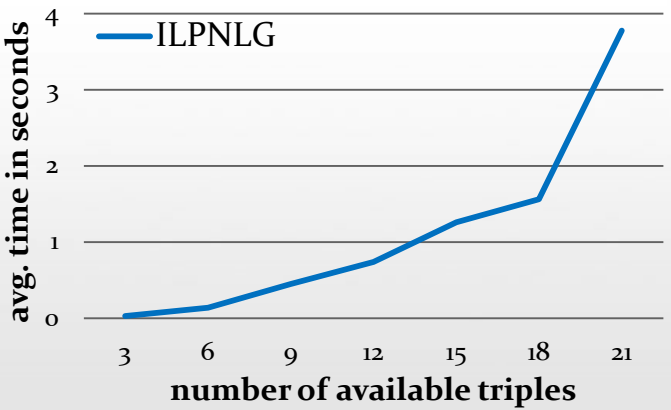
Efficiency

- **Consumer Electronics Ontology**
  - Sample of **30 individuals** with an **average of 14 triples** (max of 21 triples) per individual.
  - For **ILPNLG**, more **buckets** (10) are required to express all these **triples**.
  - However, the number of **variables** in the **ILP model** grows **exponentially** to the number of **buckets** and available **triples**.

Results are averaged for **λ1 = 0.4, 0.5, 0.6.**

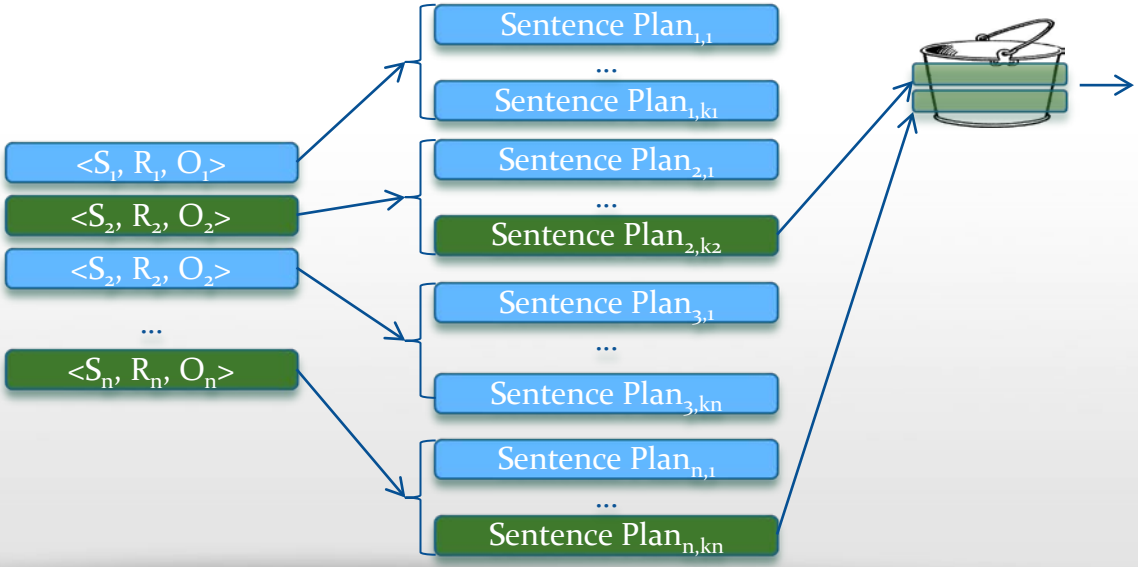
Solving times on the **Wine Ontology** averaged **0.08 sec**.





Approximation

- **ILPNLGAPPROX** system.
  - Considers each bucket (aggregated sentence) **separately**.
  - **ILP model** fills a **single bucket** to produce the **first aggregated sentence** and then we use the model again with the **remaining triples** to produce the **second aggregated sentence,** etc.



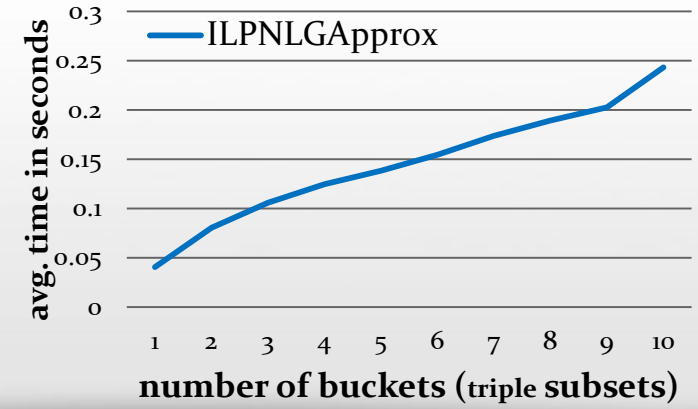
"*This is a Riesling.*" + "*It is full.*"

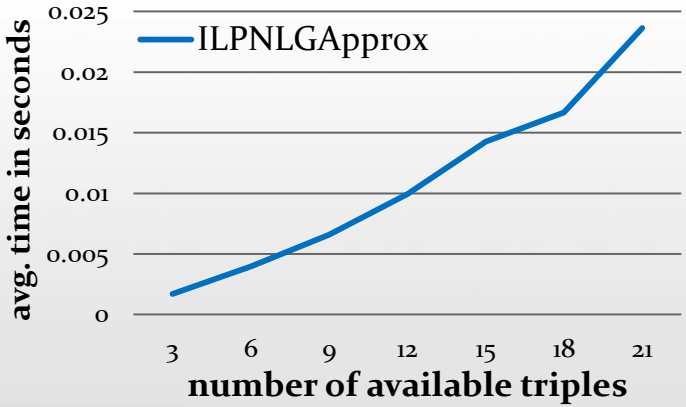
↓

"*This is a full Riesling.*"

Approximation

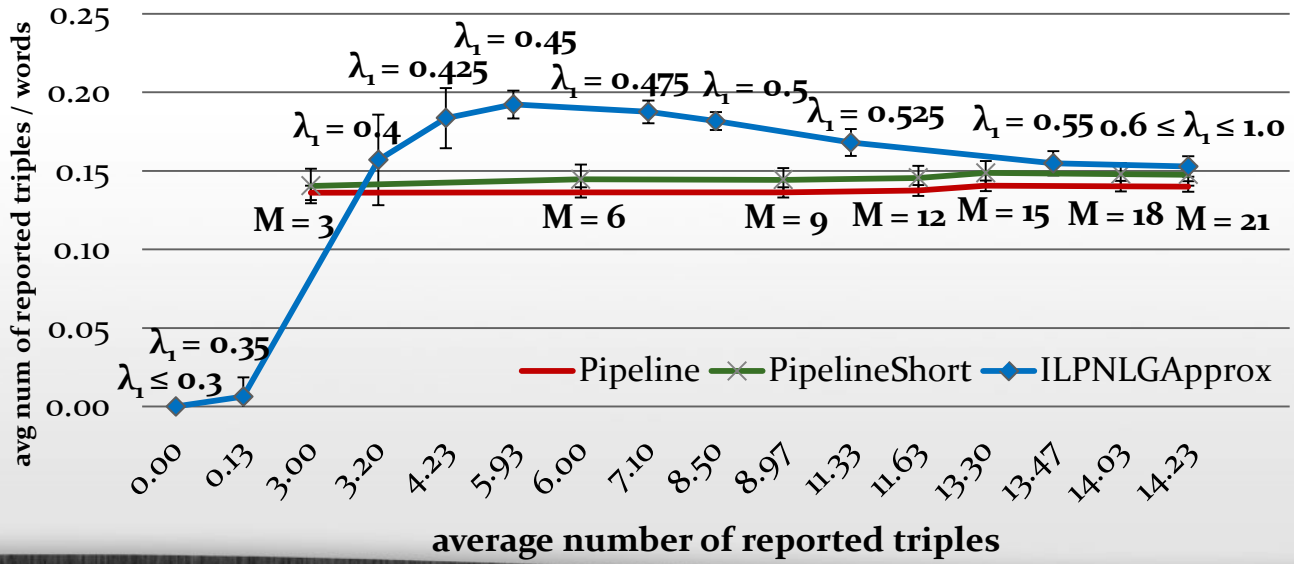
- **ILPNLGAPPROX** system.
  - Considers each bucket (aggregated sentence) **separately**.
  - **ILP model** fills a **single bucket** to produce the **first aggregated sentence** and then we use the model again with the **remaining triples** to produce the **second aggregated sentence,** etc.
  - It does not guarantee an **optimal solution,** but there was **no deterioration** of **triple/word ratios** in experiments with the **Consumer Electronics Ontology**.
  - It is **much faster** and grows approximately **linearly** to the number of **buckets** and available **triples!**





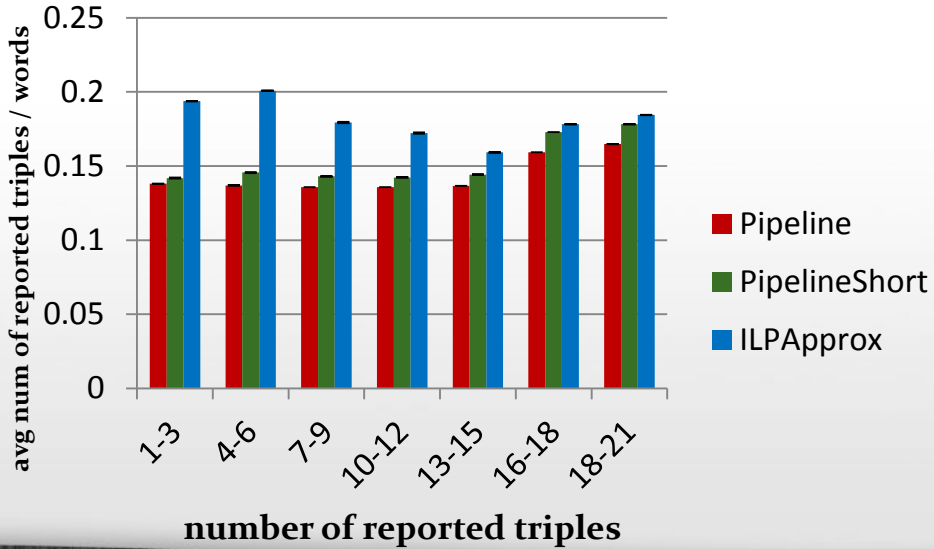
Consumer Electronics Results

- **PIPELINE:**
  - We set **M = 3,6,9,…,21** (target number of triples).
- **ILPNLGAPPROX:**
  - We generated texts for different values of **λ1 (λ2 = 1 – λ1)**.
  - **ILPNLGAPPROX** behaves similarly to **ILPNLG** in the **Wine Ontology** experiments.



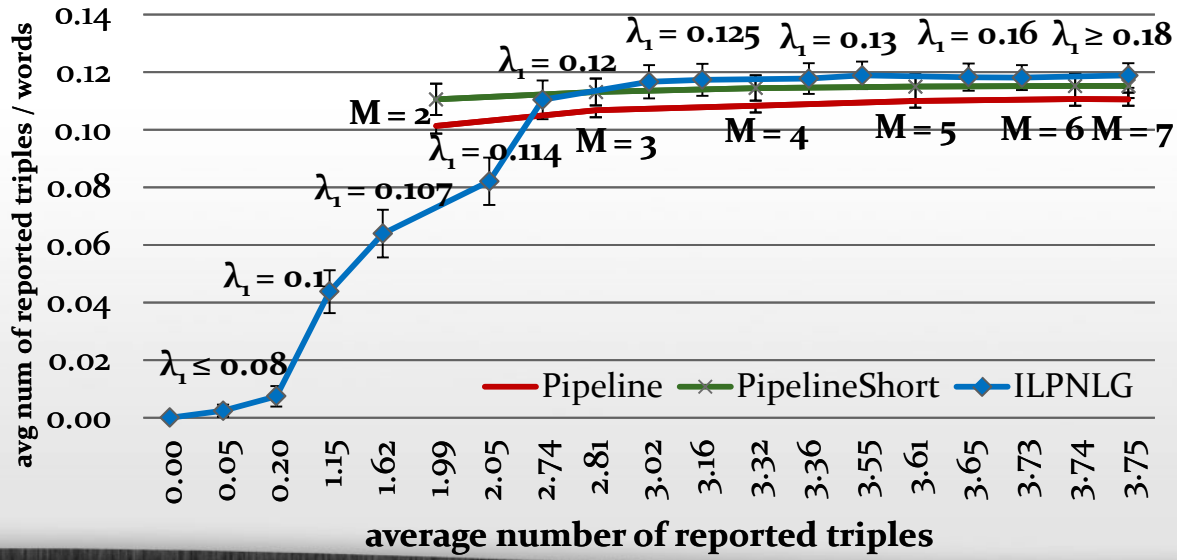
Consumer Electronics Results

- **PIPELINE:**
  - We set **M = 3,6,9,…,21** (target number of triples).
- **ILPNLGAPPROX:**
  - We generated texts for different values of **λ1 (λ2 = 1 – λ1)**.
  - **ILPNLGAPPROX** behaves similarly to **ILPNLG** in the **Wine Ontology** experiments.



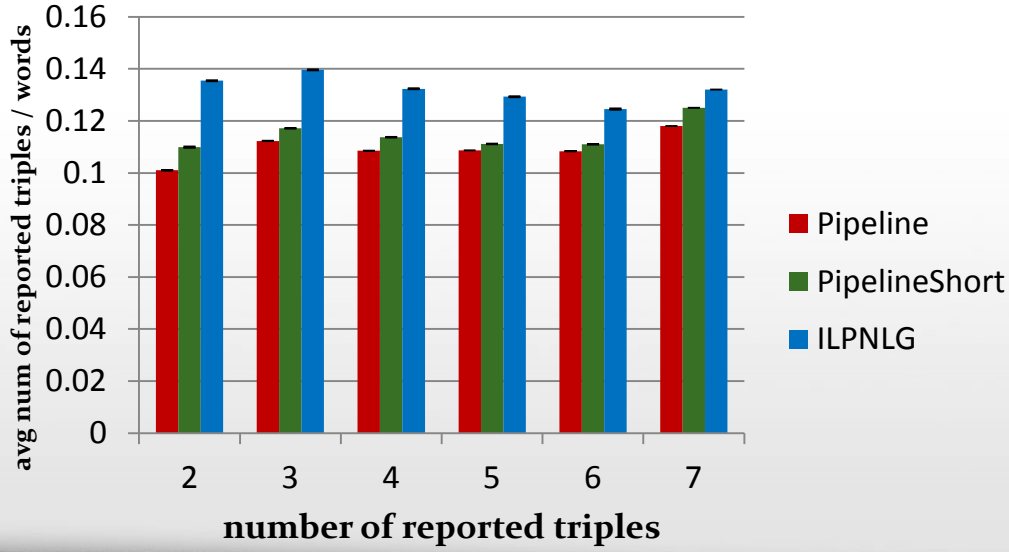
Disease Ontology Results

- Sample of **200 classes** with an **average of 4 triples** (max of 7 triples) per individual.
- **PIPELINE:**
  - We set **M = 2,3,…,7** (target number of triples).
- **ILPNLG:**
  - We generated texts for different values of **λ1 (λ2 = 1 – λ1)**.



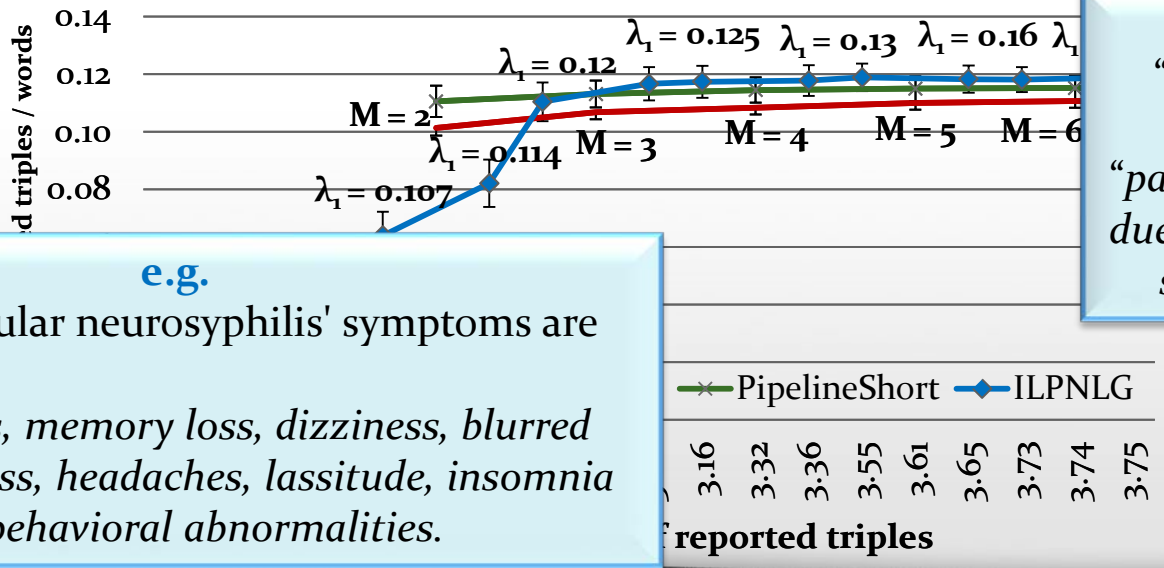
Disease Ontology Results

- Sample of **200 classes** with an **average of 4 triples** (max of 7 triples) per individual.
- **PIPELINE:**
  - We set **M = 2,3,…,7** (target number of triples).
- **ILPNLG:**
  - We generated texts for different values of **λ1 (λ2 = 1 – λ1)**.



Disease Ontology Results

- **ILPNLG** considers only the **length** of the **sentence plans** (in elements) and **not** the **length** of **NL names**.
- **ILPNLG** worked on ontologies where there was **no great variance** in **NL name** length.
- In the **Disease Ontology**, **NL names** vary from 1 to over 10 words.
  - And often there are multiple values per relation.



e.g.
Meningovascular neurosyphilis' symptoms are

neck stiffness, memory loss, dizziness, blurred vision, weakness, headaches, lassitude, insomnia and behavioral abnormalities.

e.g.
"inflammation"
vs.
"paralysis of the legs due to thrombosis of spinal arteries"

Model Extension

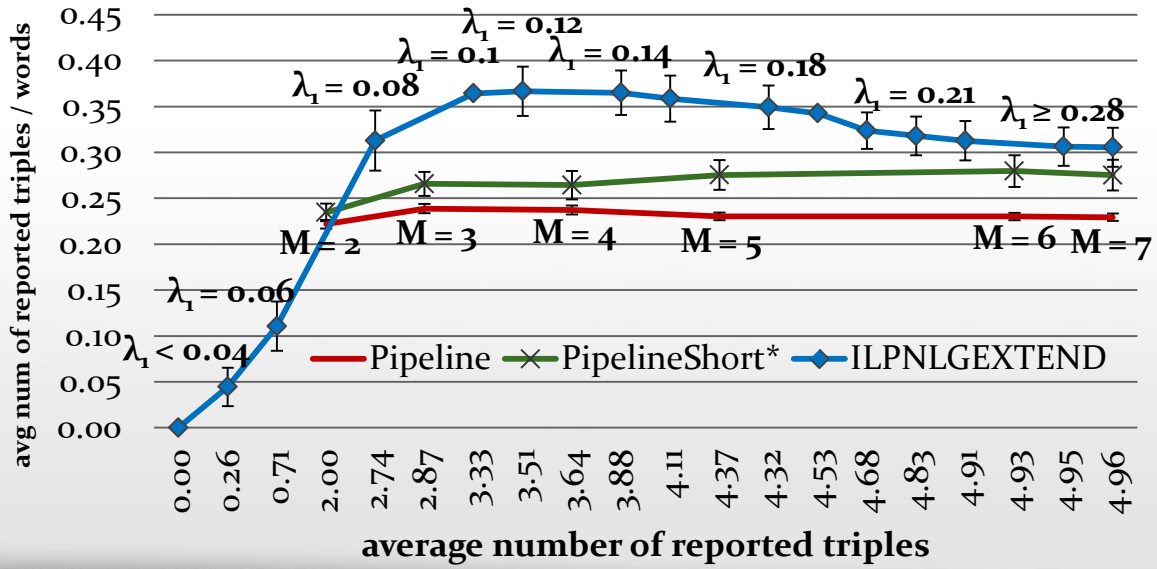
- We extend our model to consider **NL names** and how they affect **content selection**.
- Specifically the **extended model** considers:
  - the **number of NL names** appearing in each message triple,
  - the **length** of each **NL name** and **sentence plan** (in **words** instead of **elements**),
  - and an **automatic** method for **redundancy inference**.

    Kalin Cellars Semillon is a Semillon.

- **PIPELINE** now selects **randomly** among the **available sentence plans** and **available NL names**.
- **PIPELINESHORT** now selects the **shortest (in words)** sentence plan and NL name available and also attempts **to minimize redundancy**.

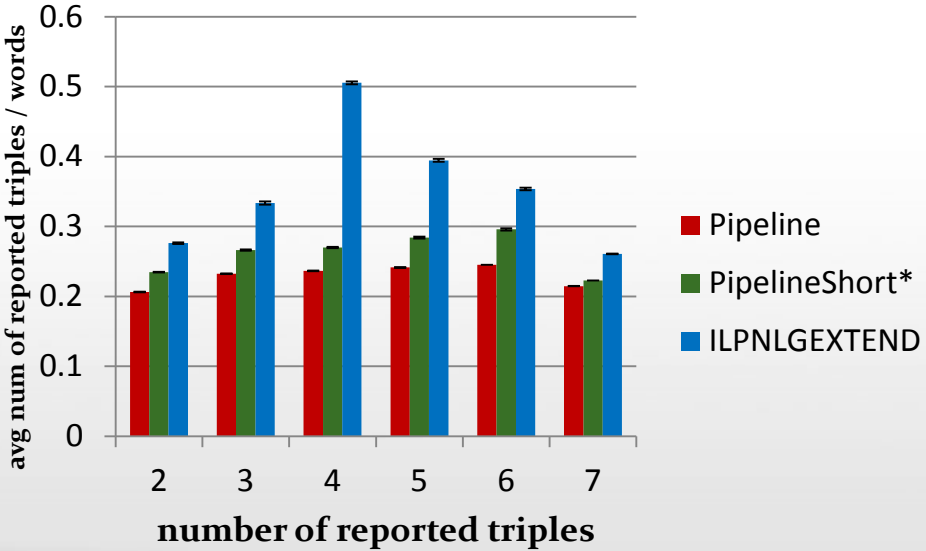
Wine Ontology Results

- The generation resources include an average of **2,8 NL Names** per individual/class (**max 5**).
- **PIPELINE:**
  - We set **M = 2,3,4,5,6** (target number of triples).
- **ILPNLGEXTEND:**
  - We generated texts for different values **λ1 (λ2 = 1 – λ1)** resulting in different number of reported triples.



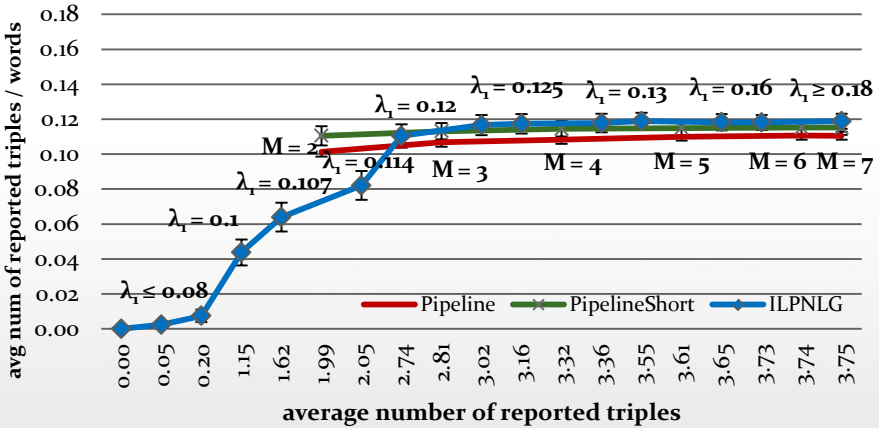
Wine Ontology Results

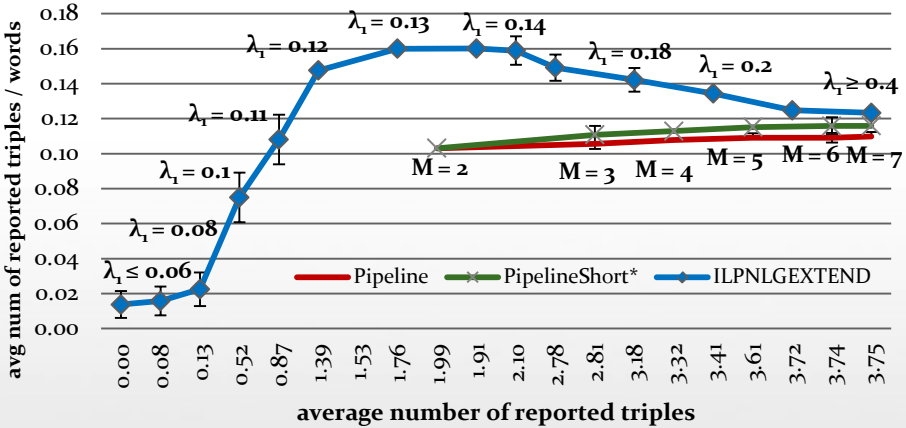
- The generation resources include an average of **2,8 NL Names** per individual/class (**max 5**).
- **PIPELINE:**
  - We set **M = 2,3,4,5,6** (target number of triples).
- **ILPNLGEXTEND:**
  - We generated texts for different values **λ1 (λ2 = 1 – λ1)** resulting in different number of reported triples.



Disease Ontology Results

- The generation resources for **Disease Ontology** include **no extra NL names**.
  - Disease names tend to be **unique** and **unvarying**.
  - However, the **extended model** considers the **length** of all **NL names included** in each message triple.



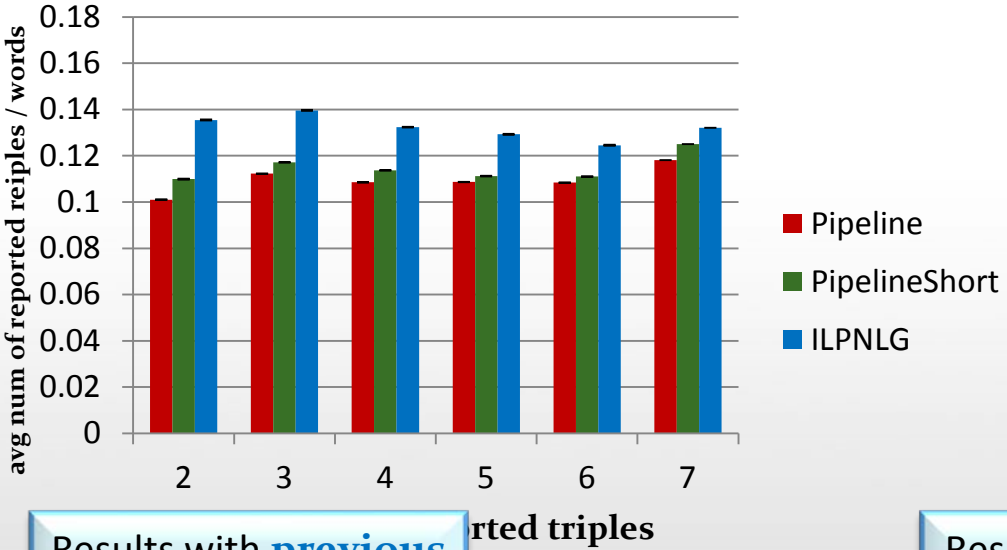


Results with **previous ILP model.**

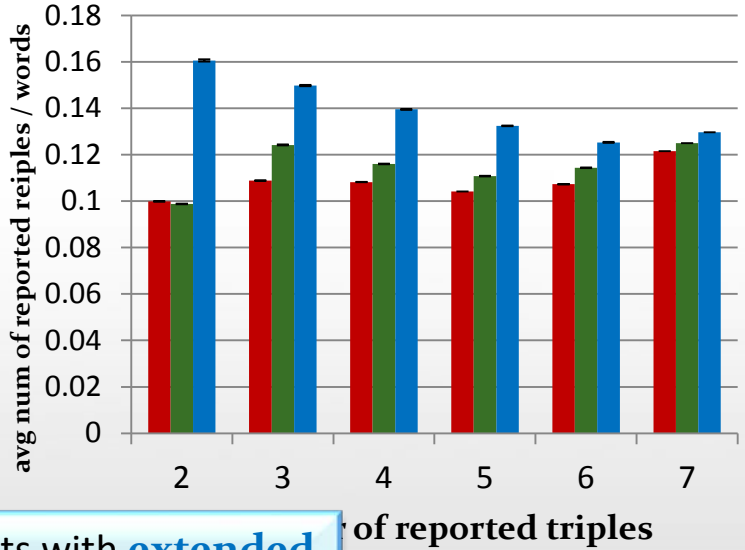
Results with **extended ILP model.**

Disease Ontology Results

- The generation resources for **Disease Ontology** include **no extra NL names**.
  - Disease names tend to be **unique** and **unvarying**.
  - However, the **extended model** considers the **length** of all **NL names included** in each message triple.



Results with **previous ILP model.**



Results with **extended ILP model.**

Related Work

- This work is the **first** to consider **content selection**, **lexicalization**, and **sentence aggregation** (and a limited form of **referring expression generation**) as an **ILP joint optimization problem** in **multi-sentence concept-to-text generation**.

- Previous **related work** either
  - employed **pipeline** architectures,
    (pipeline architectures are typical of NLG systems, NaturalOWL also previously used a pipeline)
  - used fewer or different **processing stages**,
    Marciniak and Strube (2005), Barzilay and Lapata (2005, 2006), Althaus et al. (2004), Kuznetsova et al. (2012)
  - generated **single sentences**,
    Barzilay and Lapata (2006)
  - or had very **different inputs**.
    Marciniak and Strube (2005), Kuznetsova et al. (2012)

Summary and Contribution of this Section

- We presented an **ILP model** of **content selection**, **lexicalization**, and **aggregation** that jointly considers the possible choices in all three stages to produce **more compact** texts.
- An **extended version** of the **ILP model** also considers the **lengths and number of alternative NL names** and how **to avoid redundant triples**.
- We also defined and **approximation** of our **ILP models** that generates **separately** each **sentence** of the final text.
  - It allows **longer texts** to be generated **efficiently**.
- The models have been embedded in **NaturalOWL**.

- **Experiments** with **Wine, Consumer Electronics** and **Disease ontologies** confirm that our approach leads to expressing **more triples per word**.
- **Experiments** also showed that our **ILP methods** are **efficient** enough to be used in practice.
- **Further experiments** where **human judges** evaluated the resulting texts.
  - Showed **no deterioration** in the perceived **text quality.**
  - The **ILP approach** may actually lead to texts **perceived as clearer** and **overall better,** when there are **more triples** to express.

Summary and Contribution of this Section (2)

Publications

G. Lampouras and I. Androutsopoulos,
"Using Integer Linear Programming in Concept-to-Text Generation to Produce More Compact Texts".
Proceedings of the *51st Annual Meeting of the Association for Computational Linguistics* (ACL 2013), Sofia, Bulgaria, pp. 561-566 (short papers), 2013.

G. Lampouras and I. Androutsopoulos,
"Using Integer Linear Programming for Content Selection, Lexicalization, and Aggregation to Produce Compact Texts from OWL Ontologies".
Proceedings of the *14th European Workshop on Natural Language Generation* (ENLG 2013), at the *51st Annual Meeting of the Association for Computational Linguistics* (ACL 2013), Sofia, Bulgaria, pp. 51-60, 2013.

G. Lampouras and I. Androutsopoulos,
"Producing Compact Texts with Integer Linear Programming in Concept-to-Text Generation".
Journal submission under review.

Outline

I. Natural Language Generation from Ontologies

II. Generating texts with Integer Linear Programming

III. Extracting NLG resources from the Web

Motivation

- **Natural Language Generation** systems require **domain-dependent linguistic resources** to produce **quality texts**.
  - **Authoring generation resources** can be **tedious** and **time-consuming**. The author is required to be familiar with the **domain of the ontology** in question, and with the **required resources**.
  - **NaturalOWL** already automatically generates some **tokenized sentence plans** and **NL names** when there are **no authored** resources available, but their quality is lacking.

- The **ablation experiments of the first section** showed that the **linguistic resources** with the most impact on the **quality** of the generated texts are **sentence plans** and **NL names**, followed by ordering and structure information.

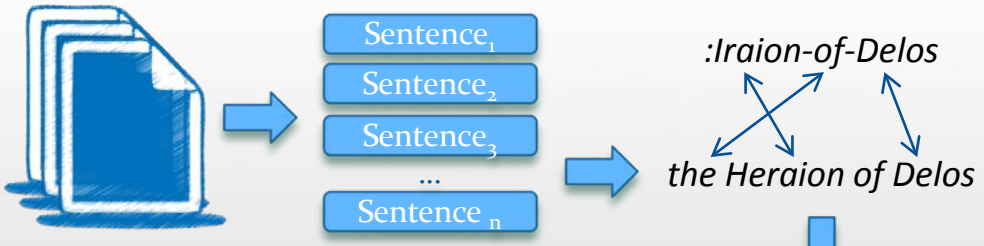
- In this section we aim to **automatically** or **semi-automatically** extract quality **sentence plans** and **NL names** from the Web.

NL Name Extraction

- For every **individual or class**, one or more template-like **NL Names** should be specified to indicate how that **individual or class** can be **expressed in the text**.
- An **NL Name** is a **sequence of slots**, along with instructions specifying how to fill them in.

NL name for :Iraion-of-Delos

$[]^{1}_{\text{article, def, agr = 2}}$ $[\textit{Heraion}]^{2}_{\text{headnoun, sing, nom}}$ $[\textit{of}]^{3}_{\text{prep}}$ $[\textit{Delos}]^{4}_{\text{noun, sing, nom}}$



$[]_{\text{article, def, agr = 2}}$ $[\textit{Heraion}]_{\text{headnoun, sing, nom}}$ $[\textit{of}]_{\text{prep}}$ $[\textit{Delos}]_{\text{noun, sing, nom}}$

Sentence Plan Extraction

- For every **relation**, one or more template-like **sentence plans** should be specified to indicate how **message triples** involving that relation can be **expressed in the text**.
- A **sentence plan** is a **sequence of slots**, along with instructions specifying how to fill them in.

Sentence plan for **:madeFrom** e.g. *<S, :madeFrom, O>*

$[ref(S)]^{1}_{nom}$ $[make]^{2}_{verb,\ passive,\ present,\ agr=1,\ polarity=+}$ $[from]^{3}_{prep}$ $[ref\ (O)]^{4}_{acc}$

- The purpose of our method is to **identify the best few** (up to 5) **candidate sentence plans**.
  - In a **fully automatic** scenario, the **candidate sentence plan** that the method considers **best** is then used.
  - In a **semi-automatic** scenario, the **top 5 candidate sentence plans** are shown to a **human author**, who picks the best one.
    - This is much **easier** and much **faster than manually authoring** sentence plans.

Seed Name Pairs

- Suppose we want to produce **sentence plans** for relation **:madeFrom**.

:Wine

:Grape

<:StEmilion, :madeFrom, :cabernetSauvignonGrape>
<:Semillon, :madeFrom, :semillonGrape>
…

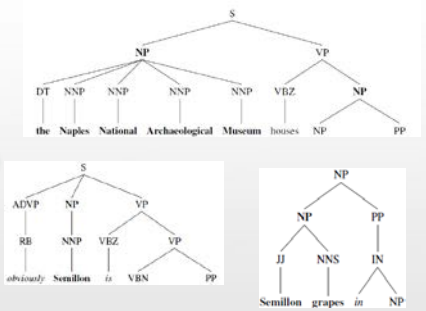
<:Wine, :madeFrom, :cabernetSauvignonGrape>
<:StEmilion, :madeFrom, :Grape>
<:Wine, :madeFrom, :Grape>
…

<NLName(S), NLName(O)>

<"St. Emilion", "Cabernet Sauvignon grape">
<"Semillon", "Semillon grape">
<"a wine", "Cabernet Sauvignon grape">
<"St. Emilion", "a grape">
<"a wine", "a grape">
…

Seed Name pairs

Sentence$_1$
Sentence$_2$
Sentence$_3$
…
Sentence $_n$



Template Extraction

<“St. Emilion”, “Cabernet Sauvignon grape”>
<“Semillon”, “Semillon grape”>
<“a wine”, “Cabernet Sauvignon grape”>
<“St. Emilion”, “a grape”>
<“a wine”, “a grape”>
…

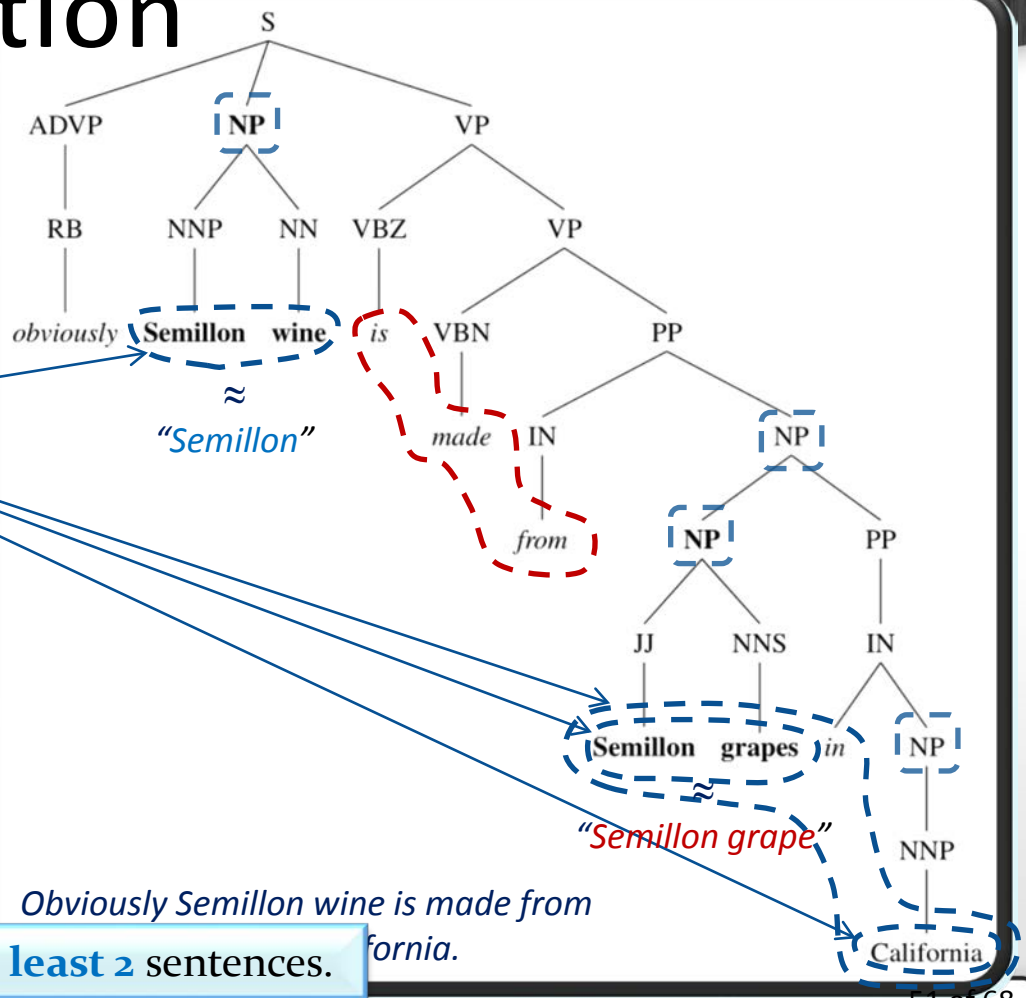
Seed Name pairs

Compare using **tf·idf weighted cosine similarity**.

S is made from O
obviously S is made from O
S is made from O in
…

Extend **templates** to produce more.

Retain **only templates** that result from **at least 2** sentences.



Turning Templates to Sentence Plans

A **template** is not yet a **sentence plan** containing the information **NaturalOWL** requires.

We **retrieve** the **POS tags** from the **parse tree** of the sentence the **template** was extracted from.

For every word we create **a corresponding slot** in the **sentence plan**.

Adjectives/**nouns**/**verbs** are looked up in the **lexicon**; if no **appropriate entries** are found, they are created.

A **dependence parser** is used to identify **slot agreements.**

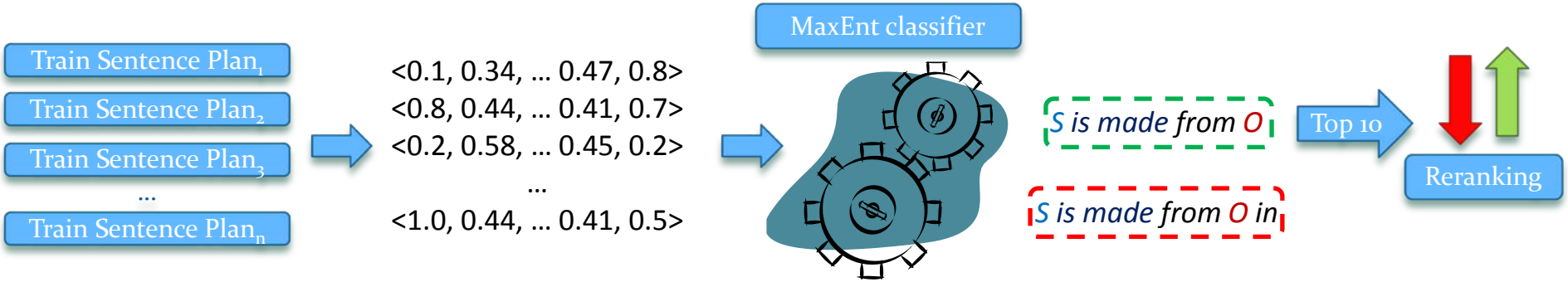
S is made from O

$S\ is_{POS=VBZ}\ made_{POS=VBN}\ from_{POS=in}\ O$

$[ref(S)]^1\ [make]^2_{\text{verb, passive, present, polarity=+}}\ [from]^3_{\text{prep}}\ [ref\ (O)]^4$

$[ref(S)]^1\ [make]^2_{\text{verb, passive, present, agr= 1,polarity=+}}\ [from]^3_{\text{prep}}\ [ref\ (O)]^4$

Scoring Candidate Sentence Plans



251-feature vector, one per candidate **sentence plan**:

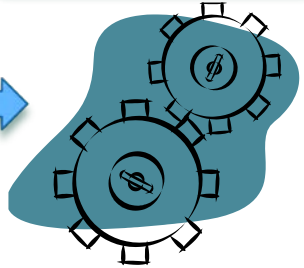
- **frequencies** of the templates, seed name pairs, sentences,
- similarities/PMI between **seed name pairs / noun phrases**,
- **grammaticality scores** of the sentence plan's generations,
- the **length** of the **sentence plan,**
- and many more – see thesis.

Scoring Candidate Sentence Plans

Sentence Plan$_1$
Sentence Plan$_2$
Sentence Plan$_3$
...
Sentence Plan$_n$

<0.1, 0.34, ... 0.47, 0.8>
<0.8, 0.44, ... 0.41, 0.7>
<0.2, 0.58, ... 0.45, 0.2>
...
<1.0, 0.44, ... 0.41, 0.5>

MaxEnt classifier



S is made from O

S is made from O in

Top 10

Reranking

<"St. Emilion", "Cabernet Sauvignon grape">
<"Semillon", "Semillon grape">
<"a wine", "Cabernet Sauvignon grape">
<"St. Emilion", "a grape">
<"a wine", "a grape">
...

Seed Name pairs

S is made from O

"St. Emilion is made from Cabernet Sauvignon grape"
"Semillon is made from Semillon grape"
"A wine is made from Cabernet Sauvignon grape"
"St. Emilion is made from a grape"
"A wine is made from a grape"
...

We **reranked** the top 10 sentence plans based on how **many sentences** returned results.

Produced Sentence Plan Inspection

- We **trained** the **classifier** on 635 **sentence plans** we manually tagged as positive/negative.
  - These sentence plans were produced from the **Wine ontology** using **manually authored NL names** in **seed name pairs** and limiting the Web documents to **Wikipedia**.

- After training, we produce and classify **sentence plans** considering **all** Web documents.
  - When classifying sentence plans **for other ontologies**, the classifier is **not retrained** on domain-specific data.

- In training and evaluation, a **sentence plan** was considered **correct** if and only if:
  - it would produce **morphologically**, **syntactically**, and **semantically correct** and **unambiguous** sentences for the corresponding relation, and
  - the **annotations** of its **slots** (e.g., POS tags, numbers, genders, agreement) were all correct.

Produced Sentence Plan Inspection

- We produced **sentence plans** with three **configurations**:
  - SP with Manual-NLN: uses the **manually authored** NL names in **seed name pairs**.
  - SP with Auto-NLN: uses the **automatically produced** NL names in **seed name pairs**.
  - BOOT: A baseline **bootstrap extraction system**.

Wine Ontology

	SP with Manual-NLN	SP with Auto-NLN	BOOT
1-in-1	**0.86** (6/7)	0.71 (5/7)	0.57 (4/7)
1-in-3	0.86 (6/7)	**1.00** (7/7)	0.71 (5/7)
1-in-5	**1.00** (7/7)	**1.00** (7/7)	0.86 (6/7)
MRR	0.89	0.86	0.68
W. 1-in-1	**0.86**	0.71	0.58
W. 1-in-3	0.86	**1.00**	0.71
W. 1-in-5	**1.00**	**1.00**	0.86
W. MRR	**0.89**	0.86	0.68

A **correct** sentence plan is ranked **highest**.

A **correct** sentence plan is in the **3** or **5 top ranked**.

Weighted scores.

Weighted by how **frequently** each **relation** appears in the **ontology statements**.

Produced Sentence Plan Inspection

- We produced **sentence plans** with three **configurations**:
  - SP with Manual-NLN: uses the **manually authored** NL names in **seed name pairs**.
  - SP with Auto-NLN: uses the **automatically produced** NL names in **seed name pairs**.
  - BOOT: A baseline **bootstrap extraction system**.

M-Piro Ontology

	SP with Manual-NLN	SP with Auto-NLN	BOOT
1-in-1	**0.67** (8/12)	**0.67** (8/12)	0.17 (2/12)
1-in-3	0.67 (8/12)	**0.75** (9/12)	0.33 (4/12)
1-in-5	0.67 (8/12)	**0.83** (10/12)	0.33 (4/12)
MRR	0.67	**0.73**	0.23
W. 1-in-1	**0.85**	0.72	0.04
W. 1-in-3	**0.85**	0.73	0.19
W. 1-in-5	0.85	**0.98**	0.45
W. MRR	**0.85**	0.79	0.22

A **correct** sentence plan is ranked **highest**.

A **correct** sentence plan is in the **3** or **5 top ranked**.

Weighted scores.

Weighted by how **frequently** each **relation** appears in the **ontology statements**.

Produced Sentence Plan Inspection

- We produced **sentence plans** with three **configurations**:
  - SP with Manual-NLN: uses the **manually authored** NL names in **seed name pairs**.
  - SP with Auto-NLN: uses the **automatically produced** NL names in **seed name pairs**.
  - BOOT: A baseline **bootstrap extraction system**.

Disease Ontology

	SP with Manual-NLN	SP with Auto-NLN	BOOT
1-in-1	0.37 (3/8)	**0.50** (4/8)	0.12 (1/8)
1-in-3	**0.87** (7/8)	**0.87** (7/8)	0.37 (3/8)
1-in-5	**1.00** (8/8)	**1.00** (8/8)	0.75 (6/8)
MRR	0.65	**0.70**	0.33
W. 1-in-1	0.16	**0.17**	0.00
W. 1-in-3	**0.89**	**0.89**	0.15
W. 1-in-5	**1.00**	**1.00**	0.85
W. MRR	**0.55**	0.48	0.22

A **correct** sentence plan is ranked **highest**.

A **correct** sentence plan is in the **3** or **5 top ranked**.

Weighted scores.

Weighted by how **frequently** each **relation** appears in the **ontology statements**.

NL Names and Sentence Plans in Generated Texts

- To investigate how the **produced NL names** and **sentence plans** affect the **generated texts**, we showed **texts** from **four configurations** of **NaturalOWL** to **six human judges**.
  - No-NLN-SP: uses **no** NL names or sentence plans (uses tokenized OWL identifiers).
  - Auto-NLN-SP: uses **top-ranked** produced NL names and produced sentence plans.
  - Semi-Auto-NLN-SP: uses NL names and sentence plans that a **human** picked from the **top five**.
  - Manual-NLN-SP: uses **manually authored** NL names and sentence plans.
- Judges were asked to **score** from **1: strong disagreement** to **5: strong agreement**.

Wine Ontology

Criteria	No-NLN-SP	Auto-NLN-SP	Semi-Auto-NLN-SP	Manual-NLN-SP
Sentence fluency	2.77	3.90	**4.67**[1]	**4.81**[1]
Clarity	3.57	2.79	**4.87**[1]	**4.93**[1]
Semantic correctness	3.57	2.86	**4.68**[1]	**4.97**[1]
Non-redundancy	3.20[1]	3.47[1]	**4.57**[2]	**4.79**[2]

We performed Analysis of Variance (ANOVA) and post-hoc Tukey tests to check for statistically significant differences (a = 0.05). A post-hoc power analysis of the ANOVA values resulted in power values equal to 1.00.

NL Names and Sentence Plans in Generated Texts

- To investigate how the **produced NL names** and **sentence plans** affect the **generated texts**, we showed **texts** from **four configurations** of **NaturalOWL** to **six human judges**.
  - No-NLN-SP: uses **no** NL names or sentence plans (uses tokenized OWL identifiers)
  - Auto-NLN-SP: uses **top-ranked** produced NL names and produced sentence plans.
  - Semi-Auto-NLN-SP: uses NL names and sentence plans that a **human** picked from the **top five**.
  - Manual-NLN-SP: uses **manually authored** NL names and sentence plans.
- Judges were asked to **score** from **1: strong disagreement** to **5: strong agreement**.

M-Piro Ontology

Criteria	No-NLN-SP	Auto-NLN-SP	Semi-Auto-NLN-SP	Manual-NLN-SP
Sentence fluency	2.10	4.33	**4.73**[1]	**4.96**[1]
Clarity	3.02[1]	3.49[1]	4.43	**5.00**
Semantic correctness	3.33[1]	2.90[1]	3.96	**5.00**
Non-redundancy	2.67	4.16[1]	**4.59**[1,2]	**5.00**[2]

We performed Analysis of Variance (ANOVA) and post-hoc Tukey tests to check for statistically significant differences (a = 0.05). A post-hoc power analysis of the ANOVA values resulted in power values equal to 1.00.

NL Names and Sentence Plans in Generated Texts

- To investigate how the **produced NL names** and **sentence plans** affect the **generated texts**, we showed **texts** from **four configurations** of **NaturalOWL** to **six human judges**.
  - No-NLN-SP: uses **no** NL names or sentence plans (uses tokenized OWL identifiers)
  - Auto-NLN-SP: uses **top-ranked** produced NL names and produced sentence plans.
  - Semi-Auto-NLN-SP: uses NL names and sentence plans that a **human** picked from the **top five**.
  - Manual-NLN-SP: uses **manually authored** NL names and sentence plans.
- Judges were asked to **score** from **1: strong disagreement** to **5: strong agreement**.

Disease Ontology

Criteria	No-NLN-SP	Auto-NLN-SP	Semi-Auto-NLN-SP	Manual-NLN-SP
Sentence fluency	3.20[1]	3.00[1]	**4.33**[2]	**4.86**[2]
Clarity	3.77	2.13	**4.47**[1]	**4.90**[1]
Semantic correctness	3.77[1]	2.43	**4.20**[1,2]	**4.93**[2]
Non-redundancy	**4.20**[1,2]	3.43[1]	**4.37**[2]	**4.70**[2]

We performed Analysis of Variance (ANOVA) and post-hoc Tukey tests to check for statistically significant differences (a = 0.05). A post-hoc power analysis of the ANOVA values resulted in power values equal to 1.00.

Related Work

- Most **related work** (Ratnaparkhi (2000), Angeli et al. (2010), Wong and Mooney (2006; 2007), Lu et al. (2009), Konstas and Lapata (2012)) requires a **parallel training corpus** of **texts** and their **semantic representations** to acquire similar linguistic resources.
  - Corpora of this kind are **difficult** to obtain.

- They also assume that **English names** of the various **individuals and classes** are **already available** in the **semantic representations**, or that they can be **obtained from their identifiers**.
  - We have proposed methods to **automatically produce** appropriate **NL names** and shown how they affect the perceived **quality of generated texts**.

- The fields of **information extraction**, **paraphrase** and **textual entailment recognition** are also relevant, with many methods concerning **template extraction**.
  - However, our **sentence plans** and **NL names** are **not simple templates**.
  - Our goal is to obtain a **single sentence plan** per relation that leads to **high quality texts**, not **all alternative natural language expressions** that convey a particular relation.

Summary and Contribution of this Section

- We presented **methods** to **automatically** or **semi-automatically** extract from the Web **sentence plans** and **NL names**.
  - We also developed a method to **avoid redundant information** in the texts.

- Experiments show that **linguistic resources** generated by the **semi-automatic method**, with **minimal human involvement**, produce texts almost as well as **manually authored resources**.
  - And much better than texts produced by resources extracted from **identifiers** of the **ontology**.

- The **manual effort** to **semi-automatically construct** the resources was a **few hours** per ontology.
  - **Manually authoring** the same resources is typically a matter of **several days**.
  - No **familiarity with OWL** and the **inner workings of NaturalOWL** are needed for an end-user to produce linguistic resources using these methods.

- The methods have been embedded in **NaturalOWL**.
  - **NaturalOWL's** processing stages are typical of **NLG systems**.

Summary and Contribution of this Section (2)

Publications

G. Lampouras and I. Androutsopoulos,

"Extracting Linguistic Resources from the Web for Concept-to-Text Generation".
Journal submission under review.

Summary and Contribution of the Thesis (1)

- We presented **NaturalOWL 2.0,** an open-source **Natural Language Generation** system that produces **fluent and coherent** multi-sentence **English** and **Greek** texts describing classes and individuals of **OWL ontologies**.
  - It is the first complete, general-purpose **NLG system for OWL ontologies**.
  - The **domain-dependent linguistic resources** are also modeled as **OWL ontologies**.

- **Experiments** with **the Wine, Consumer Electronics and Disease ontologies** showed that:
  - **NaturalOWL**, compared to simpler **OWL verbalizers**, can produce **significantly better** natural language descriptions of classes and individuals, provided that appropriate domain-dependent generation resources are available,
  - the descriptions can be **generated in more than one languages**, again provided that appropriate resources are available.

- **Ablation experiments** examined the contribution of the **various components** of the **generation pipeline** and the **domain-dependent linguistic resources** to the **quality** of the generated texts. **Sentence plans** and **NL names** seem to have the greatest effect, followed by ordering resources.

Summary and Contribution of the Thesis (2)

- We presented an **ILP model** of **content selection**, **lexicalization**, and **aggregation** that jointly considers the possible choices in all three stages to produce **more compact** texts.
- An **extended version** of the **ILP model** also considers the **lengths and number of alternative NL names** and how **to avoid redundant triples**.
- We also defined and **approximation** of our **ILP models** that generates **separately** each **sentence** of the final text.
  - It allows **longer texts** to be generated **efficiently**.

- **Experiments** with **Wine, Consumer Electronics** and **Disease ontologies** confirm that our approach leads to expressing **more triples per word**.
- **Experiments** also showed that our **ILP methods** are **efficient** enough to be used in practice.
- **Further experiments** where **human judges** evaluated the resulting texts.
  - Showed **no deterioration** in the perceived **text quality.**
  - The **ILP approach** may actually lead to texts **perceived as clearer** and **overall better,** when there are **more triples** to express.

Summary and Contribution of the Thesis (3)

- We presented **methods** to **automatically** or **semi-automatically** extract from the Web **sentence plans** and **NL names**.
  - We also developed a method to **avoid redundant information** in the texts.

- Experiments show that **linguistic resources** generated by the **semi-automatic method**, with **minimal human involvement**, produce texts almost as well as **manually authored resources**.
  - And much better than texts produced by resources extracted from **identifiers** of the **ontology**.
- The **manual effort** to **semi-automatically construct** the resources was a **few hours** per ontology.
  - **Manually authoring** the same resources is typically a matter of **several days**.
  - No **familiarity with OWL** and the **inner workings of NaturalOWL** are needed for an end-user to produce linguistic resources using these methods.

- All **systems** developed throughout this thesis are **publicly available** and **open-source**.
- The **domain-dependent linguistic resources** that were **authored** and **automatically generated** throughout this thesis, are also **publicly available**.

Publications

I. Androutsopoulos, G. Lampouras and D. Galanis, "Generating Natural Language Descriptions from OWL Ontologies: the NaturalOWL System". *Journal of Artificial Intelligence Research*, 48:671-715, 2013.

G. Lampouras and I. Androutsopoulos, "Using Integer Linear Programming for Content Selection, Lexicalization, and Aggregation to Produce Compact Texts from OWL Ontologies". Proceedings of the *14th European Workshop on Natural Language Generation* (ENLG 2013), at the *51st Annual Meeting of the Association for Computational Linguistics* (ACL 2013), Sofia, Bulgaria, pp. 51-60, 2013.

G. Lampouras and I. Androutsopoulos, "Using Integer Linear Programming in Concept-to-Text Generation to Produce More Compact Texts". Proceedings of the *51st Annual Meeting of the Association for Computational Linguistics* (ACL 2013), Sofia, Bulgaria, pp. 561-566 (short papers), 2013.

G. Lampouras and I. Androutsopoulos, "Extracting Linguistic Resources from the Web for Concept-to-Text Generation". Journal submission under review.

G. Lampouras and I. Androutsopoulos, "Producing Compact Texts with Integer Linear Programming in Concept-to-Text Generation". Journal submission under review.

D. Galanis, G. Lampouras and I. Androutsopoulos, "Extractive Multi-Document Summarization with Integer Linear Programming and Support Vector Regression". Proceedings of the *24th International Conference on Computational Linguistics* (COLING 2012), Mumbai, India, 2012.

D. Galanis, G. Karakatsiotis, G. Lampouras and I. Androutsopoulos, "An Open-Source Natural Language Generator for OWL Ontologies and its Use in Protege and Second Life". System demonstration, *12th Conference of the European Chapter* of the *Association for Computational Linguistics* (EACL 2009), Athens, Greece, 2009.

S. Konstantopoulos, A. Tegos, D. Bilidas, I. Androutsopoulos, G. Lampouras, P. Malakasiotis, C. Matheson and O. Deroo, "Adaptive Natural Language Interaction". System demonstration, *12th Conference of the European Chapter* of the *Association for Computational Linguistics* (EACL 2009), Athens, Greece, 2009.

G. Karakatsiotis, D. Galanis, G. Lampouras and I. Androutsopoulos, "NaturalOWL: Generating Texts from OWL Ontologies in Protege and in Second Life". System demonstration, *18th European Conference on Artificial Intelligence* (ECAI 2009), Patras, Greece, 2008.

G. Lampouras and I. Androutsopoulos, "Finding Short Definitions of Terms on Web Pages". Proceedings of the *2009 Conference on Empirical Methods on Natural Language Processing* (EMNLP 2009 at ACL/IJCNLP 2009), Suntec, Singapore, 2009.

Thank you!

Questions?