Named Entity Detection and Entity Linking in the Context of Semantic Web

Exploring the ambiguity question.

Eric Charton, Ph.D., Msc.

POLYTECHNIQUE MONTRÉAL
LE GÉNIE EN PREMIÈRE CLASSE

Understanding language with computers?

Understanding language with computers !

To build high level natural language understanding component, you need to know the exact identity and meaning of a word sequence.

Understanding language with computers !

To build high level natural language understanding component, you need to know the exact identity and meaning of a word sequence.

Example: analyzing a sentence with an hotel and its description.

"I want to book a room in an hotel located in the heart of Paris, just a stone's throw from the Eiffel Tower"

Understanding language with computers !

To build high level natural language understanding component, you need to know the exact identity and meaning of a word sequence.

Example: analyzing a sentence with an hotel and its description.

"I want to book a room in an hotel located in the heart of Paris, just a stone's throw from the Eiffel Tower"

- Finding a set of valid hotel references will depend on the identification of exact city reference.

Definition

Semantic annotation is the identification of a **textual entity** with its meaning through a **link** to a **graph** description.

The Semantic Annotation task consists in establishing a **URI link** between the **text mention** and a **graph on the Semantic Web**: this is **Entity Linking**.

The ambiguity question

The essential problem of semantic annotation is related to the ambiguous nature of natural language.

We can use various level of information to understand a sentence and to disambiguate it.

From the letter sequence to the meaning of the word

1. Typographical and Lexical information: capital letter inside a sentence defines a proper name (ex: **Montreal**), punctuation separates units.
2. Part-Of-Speech tagging : grammatical information. Is the word a **Verb** an **Adjective**, etc.?
3. Boundary and surface form: a group of words defines a lexical unit: ex **RMS Titanic**.
4. Named Entity (**NE**) : the semantic class of a lexical unit. Is it a **PERSon**? A **PRODuct**? any class ?
5. Entity meaning : the exact and unique ontological knowledge related to the word or the lexical unit.

Named Entity can be seen as a first level of semantic annotation:

Specific nature of Named Entity

- The NE detection task consists in assigning a class label to a lexical unit: pers.hum, loc.fac, org.com.
- The class label is unique and cannot be used to define semantic attributes of NE.
- Named Entity class is not restricted: biological entity, object category...

Example

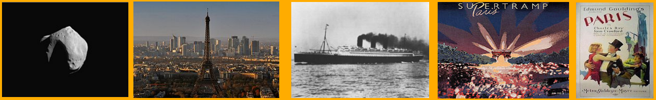
The Novotel Paris Tour Eiffel has 764 rooms and is located in the heart of Paris, overlooking the Seine and just a stone's throw from the Eiffel Tower

Named entity detection

Example: Class disambiguation

The Novotel Paris Tour Eiffel has 764 rooms and is located in the heart of $\boxed{\text{Paris}}$, overlooking the Seine and just a stone's throw from the Eiffel Tower

Named entity detection
Class selection



An asteroid (3317 Paris), a town, a ship, a movie?

Example: Final class choice

The Novotel Paris Tour Eiffel has 764 rooms and is located in the heart of Paris, overlooking the Seine and just a stone's throw from the Eiffel Tower

A city -> LOC.ADMI class

Example: But there is still an ambiguity...

The Novotel Paris Tour Eiffel has 764 rooms and is located in the heart of Paris, overlooking the Seine and just a stone's throw from the Eiffel Tower

Named entity detection
Class attribution : a city LOC.ADMI

Il n'y a qu'un seul Paris ?

... inside the class.

The Novotel Paris Tour Eiffel has 764 rooms and is located in the heart of Paris, overlooking the Seine and just a stone's throw from the Eiffel Tower

Remaining ambiguity is the main limitation of named entities for use in high level text understanding components.

Named Entity class label is limited by the tree taxonomy:

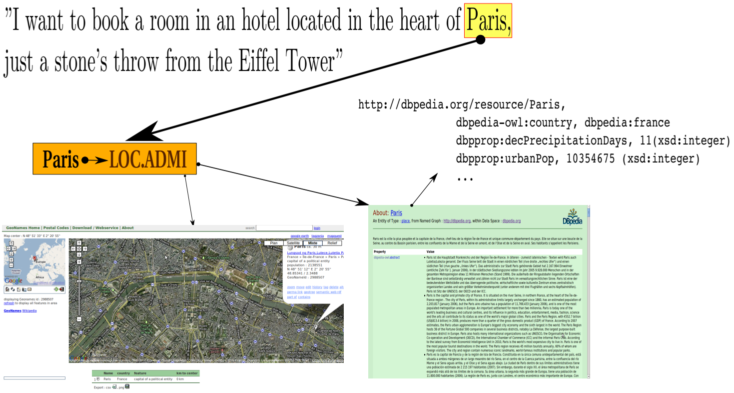
Named Entity taxonomy limitation

- Taxonomic sample: *Paris* → loc → loc.**admi** → loc.admi.**xxx**
- Semantic annotation is outside the scope of named entity taxonomy: population ? founders ? ... and which Paris ?

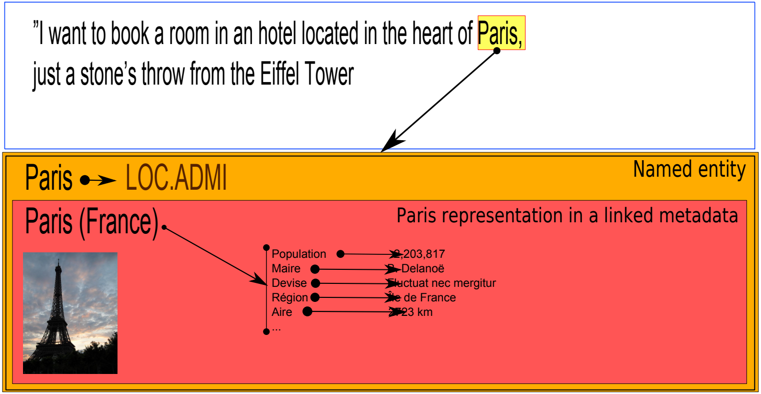
We need an external graph knowledge to introduce such information related to identity.

In modern systems, semantic knowledge is provided by the Semantic Web content through its standards.

Principle of Entity Linking is to define a link between a lexical unit and its representation on the Semantic Web.



Entity linking can solve the problem of remaining ambiguity.



But we need complementary disambiguation techniques to choose the right link.

"I want to book a room in an hotel located in the heart of Paris just a stone's throw from the Eiffel Tower".

How many Eiffel towers are located in the center of Paris with an hotel nearby?

Intuition is never a friend in NLP.

There is always a possibility of ambiguity: Semantic Annotation task is impossible to solve globally with simple finite solutions (automatons, rules). *This can be shown by experiment.*

Intuition is never a friend in NLP.

Experiment on ambiguity:

Evaluation of ambiguity on a reference corpus

Trec QA Corpus 2004.
Generic questions like:

- *"How many members were in the crew of the Challenger?"*
- *"What kind of ship is the Liberty Bell?"*
- *"When was James Dean born ?"*

Annotation of all the identifiable concepts using lexical units matched by surface forms derived from Wikipedia.

Intuition is never a friend in NLP.

Evaluation of ambiguity on Trec QA Corpus 2004.

- 350 queries
- 1126 lexical units (a unique concept defined by *n* words)
- 1076 lexical units with one ore more potential match in Wikipedia

- Over 5000 propositions for 1076 candidates
- **mean of 5.22 propositions for each lexical unit**

Intuition is never a friend in NLP.


```
<query num=15>
    <text>When was James Dean born</text>
    <match><questionmarker>When was </questionmarker><match>
    <match><questionmarker></questionmarker><match>
    <match><surfaceform lg=2>James Dean</surfaceform>
        <entity>Jimmy Dean</entity><LODuri>dbpedia.Jimmy_Dean</LODuri><NE>PERS.HUM</NE>
        <entity>James Dean (film)</entity><LODuri></LODuri><NE>prod.art</NE>
        <entity>Jamie Deen</entity><LODuri>dbpedia.Jamie_Deen</LODuri><NE>PERS.HUM</NE>
        <entity>Jim Dean (DFA)</entity><LODuri></LODuri><NE>PERS.HUM</NE>
        <entity>James Dean</entity><LODuri>dbpedia.James_Dean</LODuri><NE>PERS.HUM</NE>
        <entity>James Dean (I Wanna Know)</entity><LODuri></LODuri><NE>prod</NE>
        <entity>James Dean (song)</entity><LODuri></LODuri><NE>prod</NE>
        <entity>Pete the Cat: I Love My White Shoes</entity><LODuri>dbpedia.Pete_the_Cat:_I_Love_My_White_Shoes</
        <entity>James Deen</entity><LODuri>dbpedia.James_Deen</LODuri><NE>PERS.HUM</NE>
        <entity>Jimmy Dean (brand)</entity><LODuri></LODuri><NE>PROD</NE>
        <entity>Jimmy Dean (song)</entity><LODuri></LODuri><NE>prod</NE>
    </match>
    <match><surfaceform lg=1>born</surfaceform>
        <entity>Georgina Born</entity><LODuri>dbpedia.Georgina_Born</LODuri><NE>PERS.HUM</NE>
        <entity>Born, Saxony-Anhalt</entity><LODuri>dbpedia.Born,_Saxony-Anhalt</LODuri><NE>LOC.ADMI</NE>
        <entity>Born (crater)</entity><LODuri></LODuri><NE>LOC</NE>
        <entity>Adolf Born</entity><LODuri>dbpedia.Adolf_Born</LODuri><NE>PERS.HUM</NE>
        <entity>Brooksley Born</entity><LODuri>dbpedia.Brooksley_Born</LODuri><NE>PERS.HUM</NE>
        <entity>Mompach</entity><LODuri>dbpedia.Mompach</LODuri><NE>LOC.ADMI</NE>
        <entity>Ignaz von Born</entity><LODuri>dbpedia.Ignaz_von_Born</LODuri><NE>PERS.HUM</NE>
        <entity>Le Born, LozÃ¨re</entity><LODuri></LODuri><NE>loc.admi</NE>
        <entity>Childbirth</entity><LODuri>dbpedia.Childbirth</LODuri><NE>PERS</NE>
        <entity>Born, Luxembourg</entity><LODuri>dbpedia.Born,_Luxembourg</LODuri><NE>LOC</NE>
        <entity>Miwako Okuda</entity><LODuri>dbpedia.Miwako_Okuda</LODuri><NE>PERS.HUM</NE>
        <entity>Born auf dem DarÃŸ</entity><LODuri></LODuri><NE>loc.admi</NE>
        <entity>Born (comics)</entity><LODuri></LODuri><NE>prod.art</NE>
        <entity>Born (Netherlands)</entity><LODuri></LODuri><NE>LOC.FAC</NE>
        <entity>Le Born, Haute-Garonne</entity><LODuri>dbpedia.Le_Born,_Haute-Garonne</LODuri><NE>loc.admi</NE>
        <entity>Born (EP)</entity><LODuri></LODuri><NE>prod.art</NE>
        <entity>Max Born</entity><LODuri>dbpedia.Max_Born</LODuri><NE>PERS.HUM</NE>
        <entity>Born (album)</entity><LODuri></LODuri><NE>prod.art</NE>
    </match>
    <words>2</words><wwa>2</wwa><total matching ent>29</total matching ent><matching lod>16</matching lod>
</query>
```

Architectures for Entity Linking systems

The two steps of the Entity Linking process

1) Mention detection process

The two steps of the Entity Linking process

1) Mention detection process

- Which word or lexical unit in the sentence has to be associated with a semantic link?

The two steps of the Entity Linking process

1) Mention detection process

- Which word or lexical unit in the sentence has to be associated with a semantic link?

2)Semantic disambiguation process

- Choose from a knowledge base an instance with the right meaning according to a mention, and associate a link.

First step, mention detection (and class disambiguation)

Mention detection techniques

First step, mention detection (and class disambiguation)

Mention detection techniques

- Inference : choosing some classes of entities to target (Wikimeta system, Stanford NE system)

First step, mention detection (and class disambiguation)

Mention detection techniques

- Inference : choosing some classes of entities to target (Wikimeta system, Stanford NE system)
- Rules : using a syntactic schema to detect mentions (Xerox NE system)

First step, mention detection (and class disambiguation)

Mention detection techniques

- Inference : choosing some classes of entities to target (Wikimeta system, Stanford NE system)
- Rules : using a syntactic schema to detect mentions (Xerox NE system)
- Gazetteer : using a knowledge base of potential surface forms (DBpedia Spotlight system)

First step, mention detection (and class disambiguation)

Mention detection techniques

- Inference : choosing some classes of entities to target (Wikimeta system, Stanford NE system)
- Rules : using a syntactic schema to detect mentions (Xerox NE system)
- Gazetteer : using a knowledge base of potential surface forms (DBpedia Spotlight system)

- Hybridization of all those techniques gives the best results to prepare entity linking
(experiments of NERD Platform: `http://nerd.eurecom.fr`).

Second step, semantic disambiguation

Using a resource of potential context to compare with the entity to annotate

Second step, semantic disambiguation

Using a resource of potential context to compare with the entity to annotate

- Wikipedia provides unique descriptions for millions of semantic concepts.

Second step, semantic disambiguation

Using a resource of potential context to compare with the entity to annotate

- Wikipedia provides unique descriptions for millions of semantic concepts.
- This knowledge can be used to apply Information Retrieval (IR) algorithms.

Second step, semantic disambiguation

Using a resource of potential context to compare with the entity to annotate

- Wikipedia provides unique descriptions for millions of semantic concepts.
- This knowledge can be used to apply Information Retrieval (IR) algorithms.

IR algorithms used for disambiguation

- Vector Space Model: Wikimeta system (Cosine similarity), DBpedia Spotlight (Maximum Entropy).
- Conditional probability: (Kim system).

Second step, semantic disambiguation

Using a resource of potential context to compare with the entity to annotate

- Wikipedia provides unique descriptions for millions of semantic concepts.
- This knowledge can be used to apply Information Retrieval (IR) algorithms.

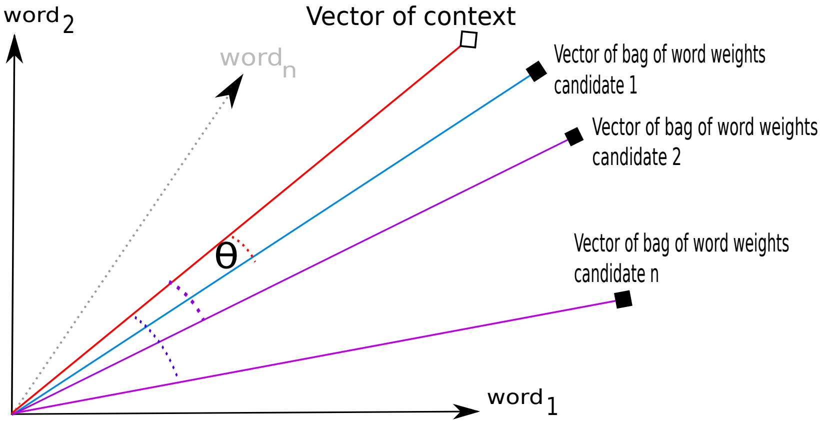
IR algorithms used for disambiguation

- Vector Space Model: Wikimeta system (Cosine similarity), DBpedia Spotlight (Maximum Entropy).
- Conditional probability: (Kim system).

- Accuracy of results can vary according to the task: mostly because of context availability.

Second step, semantic disambiguation

The Vector Space model:



Advantages of the two possible architectures

Named Entity Recognition (NER) prior to semantic disambiguation

- Robustness of detection
- Unknown or emergent concepts can receive a first level of semantic information (NE class label)
- All classes of concepts not covered by NER system are unseen.

Advantages of the two possible architectures

Named Entity Recognition (NER) prior to semantic disambiguation

- Robustness of detection
- Unknown or emergent concepts can receive a first level of semantic information (NE class label)
- All classes of concepts not covered by NER system are unseen.

Simple mention detection prior to semantic disambiguation

- Virtually any concepts can be annotated with semantic link.
- Increase of ambiguity that can minimize robustness.
- Unknown or emergent concepts non available in the gazetteer list used for mention detection are unseen.

Example of a full system architecture - Wikimeta -

Mention detection: Named Entity Recognition

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.
- Such a classifier is capable of selecting labels from a limited amount of classes (4 to 250) and only with a generic context.

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.
- Such a classifier is capable of selecting labels from a limited amount of classes (4 to 250) and only with a generic context.

(ex: A company ORG.COM) → generic contextual words → *Nyse, CAC40, Bilan, Effectifs, Shares, Revenue* etc)

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.
- Such a classifier is capable of selecting labels from a limited amount of classes (4 to 250) and only with a generic context.

(ex: A company ORG.COM) → generic contextual words → *Nyse, CAC40, Bilan, Effectifs, Shares, Revenue* etc)

Disambiguation: Semantic labeling

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.
- Such a classifier is capable of selecting labels from a limited amount of classes (4 to 250) and only with a generic context.

(ex: A company ORG.COM) → generic contextual words → *Nyse, CAC40, Bilan, Effectifs, Shares, Revenue* etc)

Disambiguation: Semantic labeling

- Unlimited quantity of potential graphs: equal to the amount of different concepts to identify.

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.
- Such a classifier is capable of selecting labels from a limited amount of classes (4 to 250) and only with a generic context.

(ex: A company ORG.COM) → generic contextual words → *Nyse, CAC40, Bilan, Effectifs, Shares, Revenue* etc)

Disambiguation: Semantic labeling

- Unlimited quantity of potential graphs: equal to the amount of different concepts to identify.
- No universal context: a unique potential context corresponds to each different concept.

Mention detection: Named Entity Recognition

- A classifier (CRF, SVM) is trained to make an inference regarding the context to extract, and to label named entities.
- Such a classifier is capable of selecting labels from a limited amount of classes (4 to 250) and only with a generic context.

(ex: A company ORG.COM) → generic contextual words → *Nyse, CAC40, Bilan, Effectifs, Shares, Revenue* etc)

Disambiguation: Semantic labeling

- Unlimited quantity of potential graphs: equal to the amount of different concepts to identify.
- No universal context: a unique potential context corresponds to each different concept.

(ex Paris (France) → personalized contextual words → *Seine, Tour Eiffel* etc).

Knowledge resource used for disambiguation

Linked Data Interface (LDI): statistical and conceptual knowledge.

Knowledge resource used for disambiguation

Linked Data Interface (LDI): statistical and conceptual knowledge.

- For each unique concept, gets all words of potential context with their TF.IDF weights.

Knowledge resource used for disambiguation

Linked Data Interface (LDI): statistical and conceptual knowledge.

- For each unique concept, gets all words of potential context with their TF.IDF weights.
- For each unique concept, includes one ore more links to the Linked Data Network.

Knowledge resource used for disambiguation

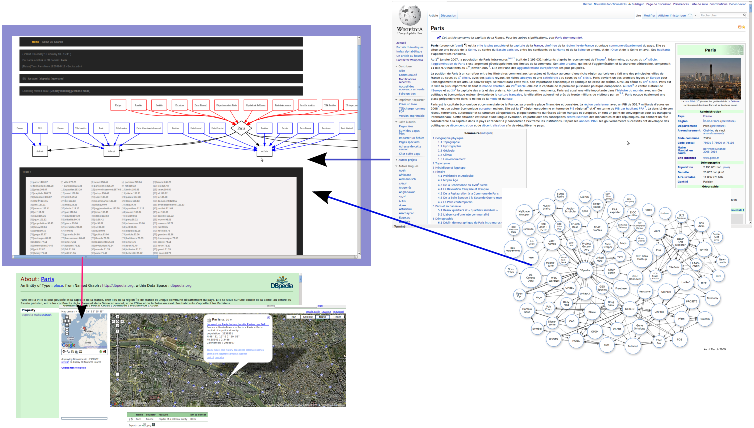
Linked Data Interface (LDI): statistical and conceptual knowledge.

- For each unique concept, gets all words of potential context with their TF.IDF weights.
- For each unique concept, includes one ore more links to the Linked Data Network.

Built from resources extracted from the Web

- Wikipedia provides 3.9M concepts with their word context.
- Each concept is associated to a set of surface forms matching lexical units.
- Correspondence tables between Wikipedia and DBpedia are used to collect links.

Linked Data Interface building



Algorithm

Named Entity detection and linking

- A NER tool detects NEs inside the text.

Algorithm

Named Entity detection and linking

- A NER tool detects NEs inside the text.
- The surface form of the NE is used to locate SE candidates in the Linked Data Interface

Algorithm

Named Entity detection and linking

- A NER tool detects NEs inside the text.
- The surface form of the NE is used to locate SE candidates in the Linked Data Interface
    - A cosine similarity measure is achieved between the context of NE and SE candidates.

Algorithm

Named Entity detection and linking

- A NER tool detects NEs inside the text.
- The surface form of the NE is used to locate SE candidates in the Linked Data Interface
  - A cosine similarity measure is achieved between the context of NE and SE candidates.
  - If more than one candidate exists (ex *Paris (France), Paris (Ontario)* ...), the best cosine score gives the best SE instance.

Algorithm

Named Entity detection and linking

- A NER tool detects NEs inside the text.
- The surface form of the NE is used to locate SE candidates in the Linked Data Interface
  - A cosine similarity measure is achieved between the context of NE and SE candidates.
  - If more than one candidate exists (ex *Paris (France), Paris (Ontario)* ...), the best cosine score gives the best SE instance.
  - A threshold value is used to reject low scored candidates (presumed wrong identification).

Algorithm

Named Entity detection and linking

- A NER tool detects NEs inside the text.
- The surface form of the NE is used to locate SE candidates in the Linked Data Interface
  - A cosine similarity measure is achieved between the context of NE and SE candidates.
  - If more than one candidate exists (ex *Paris (France), Paris (Ontario)* ...), the best cosine score gives the best SE instance.
  - A threshold value is used to reject low scored candidates (presumed wrong identification).
- Final retained corresponding instance of LDI gives the semantic link between NE and Linked Data.

Linked Data Interface (LDI)

Mention detection using a CRF classifier →

I want to book a room in an hotel located in the heart of Paris, overlooking the Seine and just a stone's throw from the Eiffel Tower

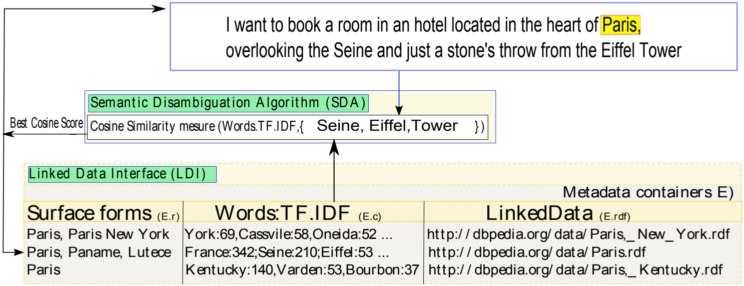
Linked Data Interface (LDI)

I want to book a room in an hotel located in the heart of Paris, overlooking the Seine and just a stone's throw from the Eiffel Tower

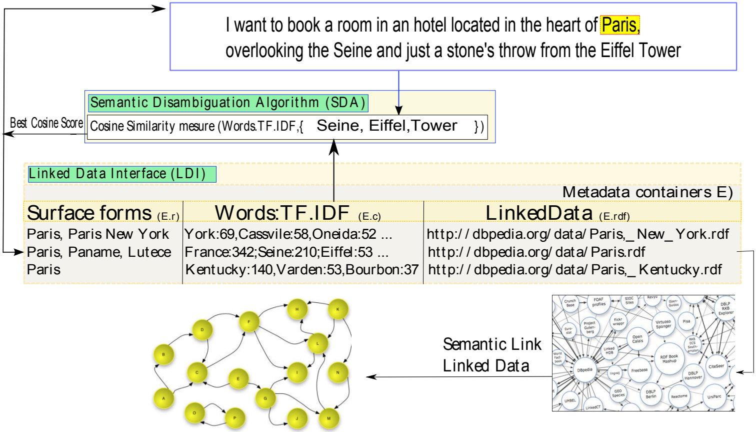
Metadata containers E)

Surface forms (E.r)	Words:TF.IDF (E.c)	LinkedData (E.rdf)
Paris, Paris New York	York:69,Cassvile:58,Oneida:52 ...	http://dbpedia.org/data/Paris,_New_York.rdf
Paris, Paname, Lutece	France:342;Seine:210;Eiffel:53 ...	http://dbpedia.org/data/Paris.rdf
Paris	Kentucky:140,Varden:53,Bourbon:37	http://dbpedia.org/data/Paris,_Kentucky.rdf

Linked Data Interface (LDI)



Linked Data Interface (LDI)



Experiments and results

Evaluation plan and method

There is no standard evaluation schema for applications like the one described here.

- We evaluated our system with an improved standard NER test corpus.
- The corpus is made of two Corpora from French and English evaluation campaigns (ESTER 2 and CoNLL 2008).
- To each NE in the corpus, we associate a standard *Linked Data* URI coming from DBpedia.

Test corpora

Word	POS	NE	Semantic Link
il	PRO:PER	UNK	
est	VER:pres	UNK	
20	NUM	TIME	
heures	NOM	TIME	
a	PRP	UNK	
Johannesburg	NAM	LOC.ADMI	`http://dbpedia.org/data/Johannesburg.rdf`

Table: Sample annotation of the French ESTER 2 NE test corpus.

Word	POS	NE	Semantic Link
Laura	NNP	PERS.HUM	*NORDF*
Colby	NNP	PERS.HUM	
in	IN	UNK	
Milan	NNP	LOC.ADMI	`http://dbpedia.org/data/Milan.rdf`

Table: Sample annotation of the English CoNLL 2008 test corpus.

Coverage of the Linked Data Interface

Each NE contained in a text document does not have necessarily a corresponding representation in LDI. The following Table shows the coverage of built metadata contained in LDI, regarding NEs contained in the test corpus.

	ESTER 2 2009 (French)			*WSJ CoNLL 2008 (English)*		
Labels	Entities in test corpus	Equivalent entities in LDI	Coverage (%)	Entities in test corpus	Equivalent entities in LDI	Coverage (%)
PERS	1096	483	44%	612	380	62%
ORG	1204	764	63%	1698	1129	66%
LOC	1218	1017	83%	739	709	96 %
PROD	59	23	39%	61	60	98 %
Total	3577	2287	64%	3110	2278	73%

Test

LDI is applied to establish a link between NEs and Linked Data network in two configuration.

- no α test mode : only the NEs covered by LDI are used.
- α test mode : all the NEs of test corpora are used and a threshold value is used.

Recall is calculated on the NE/Semantic link pairs.

Results

	French tests				English tests			
NE	[no α]	Recall	[α]	Recall	[no α]	Recall	[α]	Recall
PERS	483	0.96	1096	0.91	380	0.93	612	0.94
ORG	764	0.91	1204	0.90	1129	0.85	1608	0.86
LOC	1017	0.94	1218	0.92	709	0.84	739	0.82
PROD	23	0.60	59	0.50	60	0.85	61	0.85
Total	2287	0.93	3577	0.90	2278	0.86	3020	0.86

Table: Results of the semantic labeler applied on the ESTER 2 and WSJ CoNLL 2008 test corpora.

Conclusions

About the evolution of the semantic annotation task

The Named Entity Recognition task could be soon replaced by the Entity Linking task:

- The NE class label can be found in knowledge base using the linking.
    - One of the remaining interest of NER is its ability to discover emergent concept.
- Entity linking using surface form detection offers more possibilities of detection (common words, specific class of words like animals or organisms).
    - NER systems still offer better accuracy than simple Entity Linking systems.
    - There is work to do for improving Entity Linking robustness.

About the evolution of the semantic annotation task

Emergence of open and free structured resources like Wikipedia and Semantic Web repositories defines the nature of Semantic Annotation task:

- Semantic Web URI is a de facto standard for annotation.
- Wikipedia is the standard to build disambiguation resources.
- Wikipedia is the standard to build mention detection resources.

About the evolution of the semantic annotation task

There are still issues to solve:

- Evaluation of semantic annotation tools using standard resource is still an open research topic.
- Without enough context, semantic annotation systems still have problems.
  - The Siri (and friends) problem.
  - Semantic disambiguation in low context using reasoning is a new promising perspective of research.

Make your own experiments!

- www.wikimeta.com : semantic annotation tool with NER mention detection (free for students).
- dbpedia.org/spotlight: semantic annotation tool with Surface Form mention detection (free).
- www.nlgbase.org : semantic disambiguation resource (free CC license).
- nerd.eurecom.fr : easy to use tool to compare semantic annotation systems.

Thank you