




L'Elaborazione del Linguaggio Naturale e i Linguaggi di Markup

- dalla struttura al contenuto -





Sommario

- L'Elaborazione del Linguaggio Naturale e i linguaggi di markup
- Il markup e i linguaggi di markup
- Annotazione linguistica di testi:
 - XML (eXtensible Markup Language)
 - Esempi di testi annotati
- Definizione di ontologie:
 - RDF (Resource Description Framework)
 - OWL (Ontology Web Language)



2




L'Elaborazione del Linguaggio Naturale e i linguaggi di markup (LdM)

ELN per LdM

annotazione automatica di testi mediante tecniche di Elaborazione del Linguaggio Naturale

LdM per ELN

- annotazione linguistica di testi;
- rappresentazione del contenuto per recupero dell'informazione;



3




I linguaggi di markup

- con "markup" si intende "marcatura", "annotazione" ("to mark up" = "segnare")
- i linguaggi di markup (LdM) sono utilizzati per rappresentare un documento "marcando" alcuni suoi elementi costituenti;
- Tramite il markup è possibile strutturare un testo e permettere l'accesso ai suoi contenuti.



4

Rappresentazione di dati strutturati

- Con i LdM è possibile dare una struttura all'informazione e renderla più facilmente recuperabile ed interpretabile:

Il Signore degli Anelli,
di John Ronald Reuel Tolkien,
Bompiani.

autore titolo editore

NLP group 5

XML - eXtensible Markup Language

- XML (come HTML) è un sottoinsieme di SGML (Standard Generalized Markup Language);
- in XML tutto ciò che è compreso tra parentesi angolate è un etichetta, o *tag*;
- XML è "estensibile": consente di definire i propri tag, contrariamente ad HTML che fornisce solo tag statici (<body>, <p>, <bold>, </bold>, ecc.).

NLP group 6

Esempio di markup XML

Il Signore degli Anelli,
di John Ronald Reuel Tolkien,
Bompiani.

```
<biblioteca>
  <libro>
    <titolo>
      Il Signore degli Anelli
    </titolo>
    <autore>
      John R. R. Tolkien
    </autore>
    <editore>
      Bompiani
    </editore>
  </libro>
  ...
</biblioteca>
```

markup XML

NLP group 7

XML - DTD

- Per definire la struttura di un documento XML è necessario creare uno schema chiamato DTD (*Document Type Definition*) al quale il documento XML sarà associato. Nel nostro caso:

```
<!DOCTYPE biblioteca [
  <!ELEMENT biblioteca (libro+)>
  <!ELEMENT libro (titolo, autore+, editore)>
  <!ELEMENT titolo (#PCDATA)>
  <!ELEMENT autore (#PCDATA)>
  <!ELEMENT editore (#PCDATA)>
]>
```

uno o più elementi

uno ed un solo elemento

NLP group 8

I Linguaggi di markup per l'annotazione linguistica di testi

- E' possibile annotare un testo a diversi livelli:
 - tipografico o fonologico;
 - lessicale;
 - morfologico;
 - sintattico;
 - semantico;
 - pragmatico.

NLP group

Annotazione sintattica (1)

- es: "Frodo sedette su una roccia."

```

    <Frase>
    <Sint.Nom>
    <Nome> Frodo </Nome>
    </Sint.Nom>
    <Sint.Ver>
    <Verbo> sedette </Verbo>
    <Sint.Prep>
    <Prep> su </Prep>
    <Sint.Nom>
    <Art.> una </Art.>
    <Sostant.> roccia </Sostant.>
    </Sint.Nom>
    </Sint.Prep>
    </Sint.Ver>
    </Frase>
  
```

NLP group

Annotazione sintattica (2)

- Possibile su due tipi di informazione:
 - categorie: annotazione di stringhe in base alla **categoria sintattica** (sintagma nominale, sintagma preposizionale, ecc.)
 - dipendenze: relazioni tra elementi annotati che riflettono i loro **ruoli sintattici** (soggetto, oggetto)

NLP group

Annotazione sintattica (3)

- In base alle **categorie sintattiche**:
 - "The big dog chased the cat"

```

    S
    NP VP
    Art Adj N V NP
    The big dog chased the cat
  
```

[[NP The big dog NP] [VP chased [NP the cat NP] VP]]

NLP group

Annotazione sintattica (4)

- In base alle **dipendenze**:
 - "The big dog chased the cat"

1.	The		3
2.	big		3
3.	dog	Subj	4
4.	chased	Main V	
5.	the		6
6.	cat	Obj	4

13

Un esempio di annotazione XML: testi letterari (1)

- Obiettivo: realizzare un sistema per l'estrazione di conoscenza da un testo letterario;
- E' necessario costruire una base di conoscenza che contenga:
 - una metarappresentazione del testo;
 - una ontologia del contesto semantico.
- Testo in esame: "La Divina Commedia".

NLP group

14

Un esempio di annotazione XML: testi letterari (2)

- Il sistema deve essere capace di rispondere a domande di questo tipo:
 - Quanti e quali sono i ghibellini e quanti e quali i guelfi presenti nella Cantica Inferno della Divina Commedia?
 - Quante e quali figure della mitologia classica vengono evocate nell'Inferno?
 - Classificare l'atteggiamento di Dante verso un personaggio in base alle caratteristiche del personaggio.

NLP group

15

La Base di Conoscenza

metarappresentazione del testo

ontologia

NLP group

16

La meta-rappresentazione del testo

narrazione

```

<Narrazioni>
  <Canto> I </Canto>
  <TestoNarrazione>
    Nel mezzo del cammin...
    mi ritrovi per una selva
    ...
    Quando vidi costui nel ...
  </TestoNarrazione>
  <Inizio>
    <Verso> 1 </Verso>
    <Carattere> 1 </Carattere>
  </Inizio>
  <Fine>
    <Verso> 64 </Verso>
    <Carattere> 36 </Carattere>
  </Fine>

```

dialogo

```

<Dialogo>
  <PersonaggiDialogo>
    <Nome> Dante </Nome>
    <Nome> Virgilio </Nome>
  </PersonaggiDialogo>
  <Inizio> ... </Inizio>
  <Fine> ... </Fine>
  <SegmentiDialogo>
    <Segmento>
      <ChiParla> Dante </ChiParla>
      <AChi> Virgilio </AChi>
      <Testo>
        "Miserere di me qual che tu sii,
        od ombra od omo certal"
      </Testo>
    </Segmento>
  </SegmentiDialogo>
  <Inizio> ... </Inizio>
  <Fine> ... </Fine>
  </Segmento>
  ...

```

NLP group 17

L'ontologia: schema

- NOME
- COGNOME
- NOTO COME
- TIPO (Storico / Letterario / Creatura)
 - Storico: - EPOCA (es. Medioevo, Antichità Greca, ecc.)
 - CATEGORIA (es. Politico, Ecclesiastico, ecc.)
 - SOTTOCATEGORIA (es. Capo Ghibellino, ecc.)
 - POSIZIONE POLITICA (es. Guelfo, Ghibellino, ecc.)
 - NOTE BIOGRAFICHE (avvenimenti, ecc.)
 - NOTE
- Letterario: - AMBITO DI APPARTENENZA (es. Mitologia classica, Bibbia, ecc.)
 - CATEGORIA
 - SOTTOCATEGORIA
 - NOTE
- Creatura: - RUOLO
- SIGNIFICATO ALLEGORICO

NLP group 18

L'ontologia: XML

```

<Table>
  <Personaggi>
    <Personaggio>
      <Nome> Manente </Nome>
      <Cognome> degli Uberti </Cognome>
      <NotaCome> Farinata </NotaCome>
      <Tipo> Storico </Tipo>
      <Epoca> Medioevo </Epoca>
      <Categoria> Politica </Categoria>
      <Sottocategoria> Capo ghibellino
      </Sottocategoria>
      <PosizionePolitica> Ghibellino
      </PosizionePolitica>
      <NoteBiografiche>
        <Episodio>
          <Data> 1239 </Data>
          <Avvenimento> diventa capo del
            partito ghibellino </Avvenimento>
          <Luogo> Firenze </Luogo>
        </Episodio>
      </NoteBiografiche>
    </Personaggio>
    .....
  </Personaggi>

```

```

  <Personaggio>
    <Nome> Ulisse </Nome>
    <NotaCome> Ulisse </NotaCome>
    <Tipo> Letterario </Tipo>
    <AmbitoAppartenenza> Mitologia classica
    </AmbitoAppartenenza>
    <Categoria> Eroe greco </Categoria>
    <Caratteristica> Re di Itaca </Caratteristica>
  </Personaggio>
  <Personaggio>
    <NotaCome> Lanza </NotaCome>
    <Tipo> Creatura </Tipo>
    <Ruolo> ostacolare l'ascesa di Dante al colle
      della salvezza </Ruolo>
    <SignificatoAllegorico> lussuria
    </SignificatoAllegorico>
  </Personaggio>
  .....

```

NLP group 19

Limiti di XML

- rappresentazione della semantica limitata:
 - alla struttura (ad albero) del documento,
 - alla marcatura degli elementi (le foglie).
- XML permette di descrivere risorse e renderle leggibili da una macchina (*machine readable*) ma non comprensibili (*machine understandable*).

NLP group 20

RDF

- modello generale per fare delle "asserzioni semantiche" su oggetti Web;
- sintassi basata su XML;
- un oggetto Web (pagina, documento, ecc.) è detto *risorsa* ed è identificabile tramite un URI (*Uniform Resource Identifier*);

NLP group 21

RDF - asserzioni

- i dati sono descritti da *asserzioni*;
- un'asserzione è composta da :
 - una **risorsa** (pagina web, un intero sito, un libro, o qualunque cosa dotata di URI);
 - una **proprietà** (titolo, data, tipo, ecc.);
 - un **valore** (stringa, numero, un'altra risorsa, ecc.)

NLP group 22

RDF - esempio (1)

- Supponiamo di modellare questa asserzione:
 - "La persona di nome Marino Franzosi, email franzosi@cicap.org, è autore di <http://www.cicap.org>"

```

graph TD
    A("http://www.cicap.org/enciclop/at100301.htm") -- autore --> B("http://www.cicap.org")
    A -- nome --> C[Marino Franzosi]
    A -- email --> D[franzosi@cicap.org]
  
```

NLP group 23

RDF - esempio (2)

```

graph TD
    A("http://www.cicap.org/enciclop/at100301.htm") -- autore --> B("http://www.cicap.org")
    A -- nome --> C[Marino Franzosi]
    A -- email --> D[franzosi@cicap.org]
    
    B --- RB["risorsa (URI del sito)"]
    A --- RA["risorsa (e valore) (URI dell'autore)"]
    C --- CV["valore"]
    D --- DV["valore"]
    
    A -.->|proprietà| B
    A -.->|proprietà| C
    A -.->|proprietà| D
  
```

NLP group 24

Schemi RDF (RDF-S)

- permettono di specificare quali proprietà possono comparire nelle asserzioni RDF;
- specificano le relazioni tra proprietà;
- dichiarano classi di risorse;
- permettono di rappresentare *reti semantiche sul web* in cui:
 - nodi: URI
 - archi:
 - rdfs:Class
 - rdfs:Property
 - rdfs:subClassOf

NLP group 25

Reti semantiche sul web

- Primo passo verso creazione di ontologie

NLP group 26

OWL (Web Ontology Language)

- è un linguaggio per definire *ontologie* sul web
 - **ontologia**: rappresentazione della natura degli oggetti analizzati e delle relative interrelazioni;
- permette di esplicitare e rendere elaborabile automaticamente l'informazione contenuta nei documenti;
- è basato sulla **Logica Descrittiva**;
- è un'evoluzione del precedente DAML+OIL.

NLP group 27

OWL e il Semantic Web

Semantic Web

- permette di rappresentare proprietà e caratteristiche più complesse per classi e relazioni (OWL)
- permette la modellazione di oggetti (risorse) e la specifica di relazioni tra di essi. (RDF)
- fornisce la sintassi per documenti strutturati: nessuna rappresentazione del significato (XML)

NLP group 28

Documenti XML, RDF e OWL

- Non tutti i documenti XML sono documenti RDF, perché non rispettano certe restrizioni sintattiche;
- Analogamente non tutti i documenti RDF sono documenti OWL;
- Tutti sono comunque documenti XML sintatticamente corretti.

documenti OWL
documenti RDF
documenti XML

NLP group 29

Esempio di definizione di ontologia con OWL

- si vuol definire una ontologia di "vini", a partire dalle seguenti "classi di base" (root classes):
 - Winery (Cantina)
 - Region (Regione)
 - Consumable thing (Cosa consumabile)

NLP group 30

OWL: classi di base

```

<owl:Class rdf:ID="Winery"/>
<owl:Class rdf:ID="Region"/>
<owl:Class rdf:ID="ConsumableThing"/>
  
```

il nome della classe

sto definendo una classe OWL

assegno un nome alla classe

classe più generale

```

graph TD
  Thing[Thing]
  Winery[Winery]
  Region[Region]
  ConsumableThing[ConsumableThing]
  Winery --> Thing
  Region --> Thing
  ConsumableThing --> Thing
  
```

NLP group 31

OWL: sottoclassi

```

<owl:Class rdf:ID="PotableLiquid">
  <rdfs:subClassOf rdf:resource="#ConsumableThing" />
  ...
</owl:Class>
  
```

La classe "PotableLiquid" è definita come sottoclasse di "ConsumableThing":

```

graph TD
  ConsumableThing[ConsumableThing]
  PotableLiquid[PotableLiquid]
  PotableLiquid --> ConsumableThing
  
```

NLP group 32

OWL: individui

- Gli individui sono i membri delle classi;
- Possiamo introdurre un individuo dicendo che è membro di una certa classe:
 - es: "CentralCoastRegion" è membro della classe "Region":

```
<Region rdf:ID="CentralCoastRegion" />
```

NLP group 33

Ontologia ottenuta (1)

```

graph BT
    Thing --> ConsumableThing
    Thing --> Winery
    Thing --> Region
    ConsumableThing --> PotableLiquid
    Region --> CentralCoastRegion
  
```

NLP group 34

Classi importate

- In OWL è possibile importare ontologie, supponiamo di importare l'ontologia "food" nella quale sia stata definita la classe "Grape" (uva) in qualche modo e:
 1. Definiamo "WineGrape" come sottoclasse di "Grape" e
 2. Definiamo l'individuo "CabernetSauvignonGrape" come membro della classe "WineGrape":

```

<owl:Class rdf:ID="WineGrape">
  <rdfs:subClassOf rdf:resource="#&food;Grape" />
</owl:Class>
<WineGrape rdf:ID="CabernetSauvignonGrape" />
  
```

Classe "Grape" dell'ontologia importata "food"

NLP group 35

Ontologia ottenuta (2)

```

graph BT
    Thing --> ConsumableThing
    Thing --> Winery
    Thing --> Region
    Thing --> Grape
    ConsumableThing --> PotableLiquid
    Region --> CentralCoastRegion
    Grape --> WineGrape
    WineGrape --> CabernetSauvignonGrape
  
```

NLP group 36

1ª definizione di "vino"

```

<owl:Class rdf:ID="Wine">
  <rdfs:subClassOf rdf:resource="PotableLiquid"/>
  ...
</owl:Class>

```

La classe "Wine" è definita come sottoclasse di "PotableLiquid":

NLP group 37

Ontologia ottenuta (3)

NLP group 38

OWL: proprietà

- Una proprietà OWL è una relazione binaria;
- ogni proprietà è specificata da un dominio (*domain*) e un codominio (*range*);
- Es: definiamo la proprietà "madeFromGrape" per associare vini ad uva da frutta:

```

<owl:ObjectProperty rdf:ID="madeFromGrape">
  <rdfs:domain rdf:resource="#Wine"/>
  <rdfs:range rdf:resource="#WineGrape"/>
</owl:ObjectProperty>

```

and

nome della proprietà

dominio

codominio

- è possibile definire delle sottoproprietà, ad es. "madeFromGrape" potrebbe essere una sottoproprietà di "madeFrom".

NLP group 39

2ª definizione di "vino"

- A questo punto è possibile definire il vino come "un liquido potabile prodotto da almeno un tipo di uva da vino":

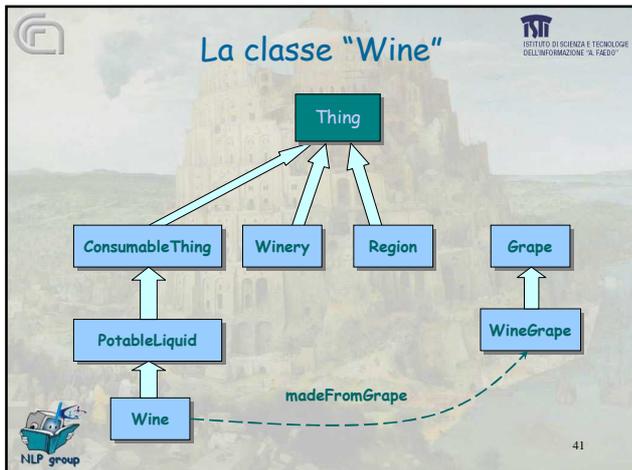
```

<owl:Class rdf:ID="Wine">
  <rdfs:subClassOf rdf:resource="PotableLiquid"/>
  and
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#madeFromGrape"/>
      <owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">
        1
      </owl:minCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  ...
</owl:Class>

```

classe senza nome (*anonymous*) che rappresenta l'insieme delle cose con almeno una proprietà "madeFromGrape"

NLP group 40



- ### Con OWL è possibile inoltre
- Dotare le proprietà di caratteristiche (transitività, simmetria, inversione, ecc.);
 - Dotare le proprietà di restrizioni (di valore, di numero, ecc.);
 - Dichiarare relazioni di equivalenza tra individui;
 - Definire classi complesse (unione, intersezione, complemento, disgiunzione, ecc.);
 - Utilizzare l'ontologia come "Base di Conoscenza" dotata di meccanismi di inferenza.
- NLP group 42