Symbolic AI Coursework - The Riddle of Steel

March 2019

1 Motivation

In this exercise we will explore the connection between Natural Language, Knowledge Representation and Inductive Logic Programming. For this purpose we will explore the domain of a fascinating material: Steel! For some motivational material on steel, please refer to this classic opening scene [1] :).

2 Setting

Imagine a researcher from BAE Systems hired you for building a smart agent for the domain of Metallurgy. Before you were called, many have failed to address this task. However, fortunately, you attended Symbolic AI: doing this job will be a breeze!

In the long run your agent will need to answer in-depth questions (Deep Semantics) about the domain. To get started you pick the wonder material of mechanical engineering (Steel) as your target subdomain and start to represent the knowledge on that subdomain.

As you are aiming towards a real-world system, which will go from Natural Language statements in text to a structured Knowledge Base, you start looking into how you can leverage regularities in natural language to build your Knowledge Base. Additionally, you investigate how existing linguistic resources, tools and semantic representation paradigms can help you to achieve this goal. Afterwards you experiment on how to use Inductive and Deductive Reasoning over structured KBs to produce new knowledge.

3 Pedagogical Reasoning or Why do I need to do this?

Knowledge Representation is fundamental for addressing more complex AI problems. The ability to build KBs at scale requires understanding the connection between natural language and semantics: how can one transition from the unstructured to the structured world (where we get intelligence and meaning).

[1] `https://www.youtube.com/watch?v=GVx4LafsvSU`

- This exercise will develop your foundations in Knowledge Representation and its connection to NL.
- You will be exposed to the principles and some of the resources that are the basis for delivering solutions to real AI problems.
- The exercise prioritises an end-to-end picture, allowing you to have a situated and less fragmented understanding of the extraction-representation-reasoning workflow. It also provides you with a comparative view of what different perspectives on semantics bring to the table and the associated representation trade-offs.
- This exercise will also expose you to the process of using AI to model a specific non-CS third-party domain. This will be present at many points in your career.

4 Rules of the Game

This task should be done individually.

5 Marking

The marking scheme is flexible as some of the questions support multiple answers. The goal is to approach the exercise from a real-life perspective (not as an overly prescribed task). Exercises 8 and 9 account for 20 marks each out of 100. Exercise 1 accounts for 0.5 and Exercise 4 for 1.5. The rest are equally distributed.

6 Submission

You will submit the this report together with the next lab report.

7 What do I need to do?

- Please organise your answers to the questions below in a single report (pdf).
- Please care about aesthetics and interpretability. Please make sure that your answers and diagrams are properly formatted. Latex is recommended for formatting your output.

8 Exercise 1 (Lexical Analysis)

Using Wikipedia as a source, let's start by looking into the lexical categories (POS Tags) behind the words of some sentences.

'Steel is an alloy of iron and carbon, and sometimes other elements. Because of its high tensile strength and low cost, it is a major component used in buildings, infrastructure, tools, ships, automobiles, machines, appliances, and weapons.'

1. POS tag the sentences above.
2. Identify the pronominal co-references.

9 Exercise 2 (C-Structures)

Using the Stanford Core NLP online [2]:

1. Plot the constituency (phrase) structure of the sentence: ' Steel is an alloy of iron and carbon, and sometimes other elements'.
2. In the class, we saw that the nominal phrasal nodes (NPs) correspond to 'molecules of meaning'. Please list them for that sentence.
3. List the coordinations within that sentence.

10 Exercise 3 (Dependencies - Exploring new territories)

In the class we did not talk about dependency parsing and dependency structures. They are a complementary perspective to constituency parsing.

1. Learn about dependency parsing (plenty of resources online). What is the difference between these two types of representation? What is emphasised by each representation?
2. Draw the dependency structure of the following sentence: ' Steel is an alloy of iron and carbon'.

11 Exercise 4 (Open IE - Semantics)

Dependency structures are a form of syntactic representation which is almost at a semantic level of representation (meaning that they are almost at a predicate-argument structure level).

[2] `http://corenlp.run/`

1. Now, let's shift gears to the semantic level and use Open Information Extraction (OpenIE) to extract the predicate-argument structure (s p o) of the following sentences (use CoreNLP for this):

   'Steel is an alloy.' 'Steel contains carbon.' 'Steel contains iron.'

2. Represent the triples above using Prolog.
3. Represent the triples above using RDF [3].
4. Formalise the aximos using Description Logics.
5. Now, analyse what happens when you get a slightly more complicated sentence into the system: 'Steel is an alloy of iron and carbon.'

12 Exercise 5 (Complex Open IE, Rhetorical Structures)

Let's push OpenIE to the next level. Please take a look on how a different system, Graphene [4], does OpenIE, by disembedding phrases and clauses (i.e. by doing sentence splitting) and assigning rhetorical relations to it. In case you need more detail, please refer to [5], [6].

Now, for the following sentence:

' As the carbon percentage content rises, steel has the ability to become harder and stronger through heat treating; however, it becomes less ductile.'

1. Identify the nucleus and the satellites.
2. Draw the diagram with the rhetorical relations.
3. Which relations are hypotactic or paratactic?
4. Write the output for that sentence using the RDF-NL notation of Graphene [7]. It is not necessary to run Graphene (but you can compare your representation with its output if you want).

13 Exercise 6 (Taxonomies, Thesauri)

Now let's go even more semantic! Let's see how WordNet can help us to bring more structured domain knowledge. Let's take a technical term, related to the

[3] `https://www.w3.org/TR/rdf11-primer/`
[4] `https://github.com/Lambda-3/Graphene`
[5] `https://github.com/Lambda-3/Graphene`
[6] `https://arxiv.org/pdf/1807.11276.pdf`
[7] `https://github.com/Lambda-3/Graphene/blob/master/wiki/files/Barack_Obama_2017_11_06.rdfnl`

crystallographic structure/phase transition of steel: *martensite* and *austenite*. Using WordNet online[8]:

1. List the WordNet glosses for these two words.
2. List the Taxonomic/Hypernym chain up to the top for these two terms.
3. List their sibling terms.
4. For the word *temper*, how many synsets do we have? Which senses are related to steel? What are its synonyms? Would you consider these perfect or near-synonyms?

14 Exercise 7 (Frame Semantics - Exploring Further)

Now let's take a look into the semantic representation of verbs using Frame Semantics. Using VerbNet as a starting point[9], for the verbs *melt* and *oxidize*:

1. Describe the frame semantics for these verbs as listed by VerbNet, PropBank and FrameNet.
2. How these representations compare?
3. Using the FrameNet semantics, draw the Conceptual Graph for the following sentence: 'At 1433 degrees, the material started to progressively melt around the edges.'

15 Exercise 8 (Ontologies - Description Logics)

Now let's use description logics to structure our domain into an ontology. You can decide if you prefer to do it using purely DL formulae or to use a supporting ontology editor [10] (recommended). You can learn to use protégé using this tutorial [11].

Convert the following natural language statements into an ontology:

```
There are many types of heat treating processes available to steel.
The most common are annealing, quenching, and tempering.

- Low-carbon steel: 0.05 to 0.30% carbon content.
```

[8] http://wordnetweb.princeton.edu/perl/webwn
[9] http://verbs.colorado.edu/verb-index/index.php
[10] https://webprotege.stanford.edu/
[11] http://mowl-power.cs.man.ac.uk/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf

```
- Medium-carbon steel: Approximately 0.3{0.6% carbon content.
Balances ductility and strength and has good wear resistance;
used for large parts, forging and automotive components.

- High-carbon steel: Approximately 0.60 to 1.00% carbon content.
Very strong, used for springs, edged tools, and high-strength wires.

- Ultra-high-carbon steel: Approximately 1.25{2.0% carbon content.
Steels that can be tempered to great hardness. Used for special purposes
like knives, axles or punches. Most steels with more than 2.5% carbon
content are made using powder metallurgy.
```

16 Exercise 9 (Inductive & Deductive Reasoning)

Steel gets different material properties (e.g. hardness, strength) depending on the concentration of carbon in the alloy and on the heating treatment performed. Depending on how you heat and cool down the material you get different crystallographic arrangements (i.e. asutenite, ferrite) which lead to different material properties.

In this exercise you will build KBs based on diagrams from Metallurgical engineering. The we will use combined inductive and deductive reasoning to answer some queries.

You should use Prolog for the deductive part and we will provide a lightweight ILP Python client for the inductive part [12].

Figure 1 shows the relationship between these variables and the regions in which different crystallographic structures are formed. This is a fundamental picture in steel metallurgy. How can we encode this into our KB?

By discretising the values of each axis in Figure 1 (representing temperature and carbon concentration respectively) we obtain the following predicates:

- HIGH_TEMPERATURE(X)
- MEDIUM_TEMPERATURE(X)
- LOW_TEMPERATURE(X)
- HIGH_CARBON(X)
- MEDIUM_CARBON(X)
- LOW_CARBON(X)

[12] https://gitlab.com/ai-systems/symbolicailabs/tree/master

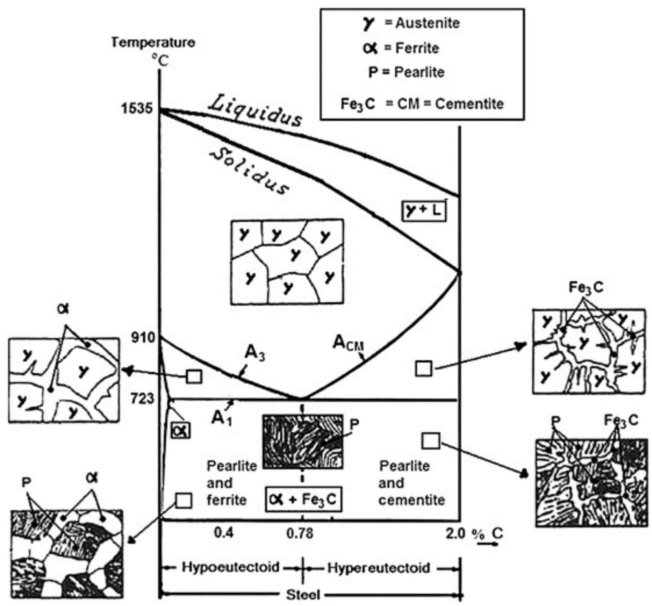

Figure 1: Crystallographic structure of steel given temperature and carbon concentration.

These predicates will constitute the building blocks to learn the definitions of each crystallographic structure.

1. The first task is using ILP to learn the definitions of each type of crystallographic structure given the sets of positive and negative examples below:

facts:

- HIGH_TEMPERATURE(1535)
- MEDIUM_TEMPERATURE(1000)
- MEDIUM_TEMPERATURE(900)
- LOW_TEMPERATURE(700)
- HIGH_CARBON(2.0)
- MEDIUM_CARBON(0.78)
- LOW_CARBON(0.4)

positive examples:

- FERRITE(900, 0.4)

- PEARLITE(700, 0.78)
- AUSTENITE(1000, 0.78)
- CEMENTITE(900, 2.0)

negative examples:

- FERRITE(700, 0.78)
- FERRITE(1535, 2)
- PEARLITE(900, 0.4)
- PEARLITE(1000, 2)
- PEARLITE(1535, 2)
- AUSTENITE(700, 0.4)
- AUSTENITE(1535, 2)
- CEMENTITE(1535, 0.4)
- CEMENTITE(700, 0.78)

You should output rules for the following structures, where X is a variable corresponding to the temperature and Y a variable corresponding to the carbon concentration:

- FERRITE(X,Y) ← ...
- PEARLITE(X,Y) ← ...
- AUSTENITE(X,Y) ← ...
- CEMENTITE(X,Y) ← ...

2. The following background knowledge describes whether a generic configuration X is liquid or solid according to the temperature:

- LIQUID(X) ← HIGH_TEMPERATURE(X)
- SOLID(X) ← MEDIUM_TEMPERATURE(X)
- SOLID(X) ← LOW_TEMPERATURE(X)

The second task requires you to perform deductive reasoning by using the learned predicates and the additional background knowledge together.

With the inductive + deductive process you should answer the following query: *Are the materials solid or liquid?*

3. Depending on the type of engineering application, different material properties are required. The third task requires you to infer the material properties of the different types of material.

By discretizing Figure 2, we can derive additional background knowledge:

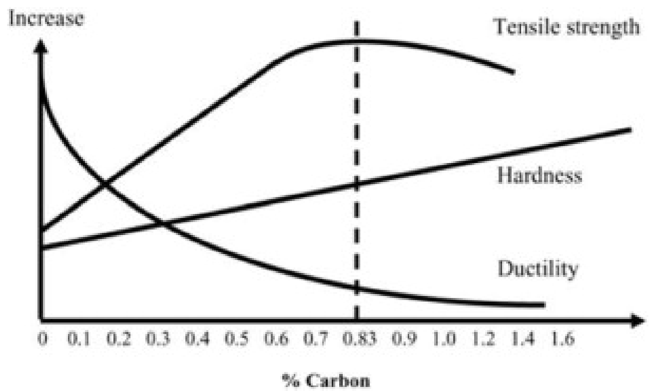

Figure 2: Material properties given carbon concentration.

- HIGH_HARDNESS(X) ← HIGH_CARBON(X)
- MEDIUM_HARDNESS(X) ← MEDIUM_CARBON(X)
- LOW_HARDNESS(X) ← LOW_CARBON(X)
- HIGH_DUCTILITY(X) ← LOW_CARBON(X)
- MEDIUM_DUCTILITY(X) ← MEDIUM_CARBON(X)
- LOW_DUCTILITY(X) ← HIGH_CARBON(X)
- HIGH_TENSILE_STRENGTH(X) ← MEDIUM_CARBON(X)
- MEDIUM_TENSILE_STRENGTH(X) ← HIGH_CARBON(X)
- LOW_TENSILE_STRENGTH(X) ← LOW_CARBON(X)

Given the additional background knowledge and the learned definitions you are required to answer the queries below. Also please provide a supporting explanation. (this again has to be done by using deduction).

- Does Ferrite have high hardness? Why?
- Is Ferrite more ductile than Cementite? Why?
- Which material has low tensile strength? Why?

Please include your KB as an appendix into the report.