CREATING IKAT: ANNOTATION TASKS

Task 1: Adding implicit knowledge between sentence pairs

In this annotation task you will be provided with pairs of sentences from the microtext corpus, a corpus of short, dense, prototypical argumentative texts written in response to a question on some potentially controversial topic (such as tuition fees, death penalty…). Each pair comes from the same text.

Your task is to help explain the connection between the pair of sentences, by providing information that makes explicit the connection between the given sentences. There may be different situations -- the connection can be obvious or not, it can be a direct link or there is more information missing. The missing information could be expressed in one or more sentences -- try to add as few sentences as possible (not in the sense of constructing a very complex sentence, but in the sense of adding the minimal amount of information to connect the two sentences in a pair). Try to make these sentences very simple -- if possible one fact/relation per sentence.

Here are a few concrete annotated examples:

1. There are direct connections between the sentences in the pair, but they are not obvious:

   e1. Schweinsteiger has been injured for several days
   e2. We urgently need a midfielder replacement for the match.

   The connection relies on knowing that *Schweinsteiger* is a football player and that his usual playing position is as *midfielder*.

   The missing information is then: **Schweinsteiger is a midfielder.**

2. Another example:

   e1. Alternative treatments should be subsidized in the same way as conventional treatments,
   e2. since both methods can lead to the prevention, mitigation or cure of an illness.

   Missing information: **Treatments are subsidized if they lead to the prevention, mitigation or cure of an illness.**

3. And another one:

   e1. Intelligence services must urgently be regulated more tightly by parliament;
   e2. this should be clear to everyone after the disclosures of Edward Snowden.

Missing information: Edward Snowden disclosed information about illegal activities of the intelligence services.

4. In the next example, two pieces of information are missing:

   e1. We were late for the party.
   e2. The engine broke down.

   Missing information: People go to a party by car. Cars have engines.

5. In this example, no information is missing:

   e1. Anti-virus programs protect the user of a computer against threats from the Internet.
   e2. That's why everyone should have an anti-virus program on their Windows computer.

   Missing information: NONE

   Lexical overlap -- such as Anti-virus programs -- Anti-virus program*, Computer--Windows-Computer* -- should not be included in the annotation because they are trivial to find (e.g. *A Windows computer is a computer*). Other than that there is no information missing in this example. To distinguish this situation from one where the annotation section for the example was left empty for other reasons, write NONE for missing information.

IMPORTANT! Usually it should be the case that you include one (or more) words of the sentence pair for filling in the missing information. Please stick to the vocabulary used in the sentences.

Examples:

- In example 1: Schweinsteiger, midfielder
- In example 2: Treatments, subsidized, prevention, mitigation or cure of an illness

Of course it is always allowed to adapt the words grammatically:

- In example 3: disclosures → disclosed

Some hints…

- Sometimes sentence pairs make only sense with a larger context, this is why for every pair we always also display the full microtext. You are always allowed to look up the context.
- If a sentence pair just doesn't make sense because of a pronoun for which you don't know the noun it is referring to, pretend as if you would know it and don't mark it as a pair that needs more context.
- If a sentence pair just doesn't make sense because of a discourse marker, ignore the discourse marker and don't mark it as a pair that needs more context.

Task 2: Annotating implicit knowledge with ConceptNet relations

Knowledge Bases are organized repositories of knowledge that promote the collection, organization and retrieval of knowledge. ConceptNet is a knowledge base containing commonsense knowledge. It consists of terms and relations between them, represented as tuples: left term + relation + right term (for example “oven, UsedFor, bake”, "car, IsA vehicle", etc.).

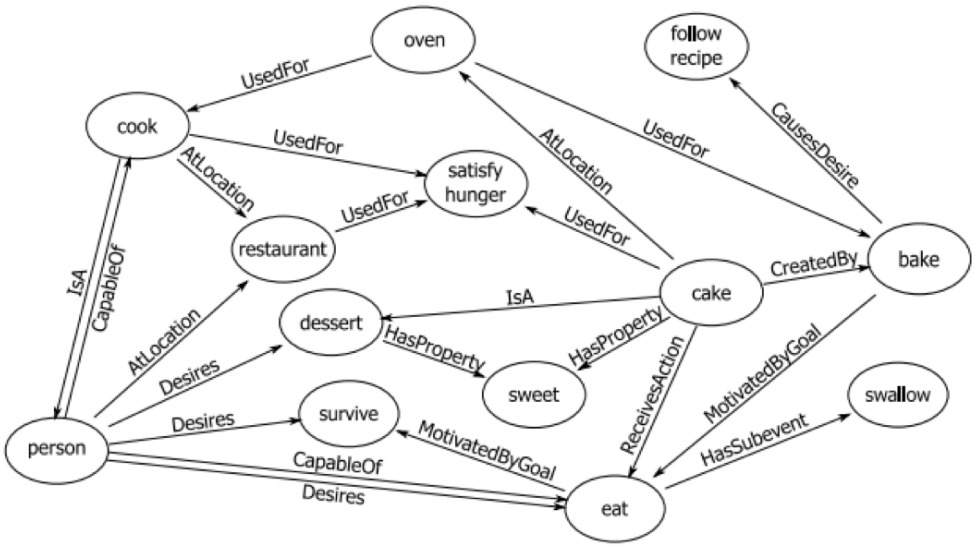

Fig. 1: Part of ConceptNet

The missing information you inserted can often be mapped to ConceptNet relations. We want you to insert the appropriate relation for the sentences that express the missing information. Please assign them to the following set of ConceptNet relations:

People go to a party by car. // UsedFor (car, going to a party)
Cars have engines. // HasA (car, engine)

This is the set of relations and their syntactic restrictions:

Relation	Example sentence pattern
IsA	NP is a kind of NP.
UsedFor	NP is used for VP.
HasA	NP has NP.
CapableOf	NP can VP.
Desires	NP wants to VP.
CreatedBy	You make NP by VP.
PartOf	NP is part of NP.
HasProperty	NP is AP.
Causes	The effect of NP|VP is NP|VP.
MadeOf	NP is made of NP.
AtLocation	Somewhere NP can be is NP.
DefinedAs	NP is defined as NP.
SymbolOf	NP represents NP.
ReceivesAction	NP can be VP (*passive*).
HasPrerequisite	Before you VP, you must VP.
MotivatedByGoal	You would VP because you want to VP.
CausesDesire	NP would make you want to VP.
HasSubevent	One of the things you do when you VP is NP|VP.
HasFirstSubevent	The first thing you do when you VP is NP|VP.
HasLastSubevent	The last thing you do when you VP is NP|VP.

AP: adjectival phrase; NP: noun phrase; VP: verb phrase. | indicates a choice between phrase types.

A nice explanation of these relations together with examples can be found here: https://github.com/commonsense/conceptnet5/wiki/Relations

Note that sometimes the relation is obvious (like in the first and the second example), and sometimes some kind of transformation is required (as in example 3 and 4).

- Example 1: Waste separation is part of environmental protection. --> PartOf (Waste separation, environmental protection.)
- Example 2: Sorting waste is still important. --> HasProperty (Sorting waste, important)
- Example 3: Officers have the power to impose fines. --> CapableOf (Officers, impose fines)
- Example 4: With a good overall grade you have a better chance of a job --> Causes (good overall grade, better chance of a job)

Some hints:

- When no ConceptNet relation fits, insert N.A.
- When you think that two ConceptNet relations fit (that shouldn't be the case very often), fill in the one you think fits better/more important for expressing the meaning of the sentence, and note the other in the comment section together with the line and column number:
    - Example: Rubbish smells, which is annoying. (Rubbish, annoying (HasProperty), alternative relation: Rubbish, smell (Causes))
- Please collect the cases where there is a relation that would be applicable, which is not part of the ConceptNet relation set! (to be annotated as NONE) in a separate sheet.
    - e.g. Requires: A side job costs time → Requires (side job, time)

Task 3: Annotating implicit knowledge with Semantic Clause Types

The missing information you inserted should also be annotated with semantic clause types. They describe the semantic types of situations (including *states, events, generics, generalizing sentences, questions*, and *imperatives*) evoked in discourse by individual clauses of text. As such, they concern the manner of presentation of content along with the conveyed information.

Below is displayed an overview of the set of semantic clause types:

Situation Entity Type	Definition	Example
State	States introduce specific properties of specific individuals to the discourse.	 *John loves cake.*
Event	Events introduce a specific event to the discourse: things that happen or happened.	 *Mike won the race.*
Generalizing Sentence	Generalizing sentences report regularities related to specific individuals.	 *Mary often feeds my cat.*
Generic Sentence	Generic sentences make statements about kinds.	 *Lions are carnivores.*
Abstract Entity	Abstract entities are a type of embedded situation, the clausal complements of verbs of knowledge or belief.	 *Mary knows that John loves cake.*

+ Questions (Who is Pete?)
+ Imperatives (Leave the house!)

A detailed description can be found here: http://www.coli.uni-saarland.de/projects/sitent/files/sitent_manual_v1.1.pdf

Please note that sometimes, a sentence contains more than one clause (when there are several main verbs), in these cases, all clauses need to be annotated:

- Example: If too much rubbish is produced, it is important to sort it. // GENERIC, GENERIC