NLP Assignment 3

Text Understanding

Anik Roy, Christ's (ar899)

December 5, 2019

Task 1

A hypothetical system would look at the key verb in the question - here, 'use'. We can find 'use' due to the xcomp dependency from the root. The 'dep' type can then show us what the object of the verb is, here 'water'. The determiner ,'what', determines the type of question, and indicates we need to look for a noun and its dependents which are modifiers.

The question *What sort of water are you advised to use?* has the answer *distilled water*, but the question answering system described above would likely return two possible answers - tap water **and** distilled water. In both relevant sentences, 'water' is dependent on a form of the verb 'use' with the direct object (dobj) type. We can take advantage of morphology [1] here to detect that 'using' is a form of 'use'. Both 'distilled' and 'tap' are dependents and modifiers of 'water' (amod/compound), as required by the question. To choose a single answer, a simple heuristic is simply choosing the latter. However, we could also look at the other dependents of 'use', i.e. 'last longer', classifying as a positive phrase, making this sentence more likely to contain the correct answer.

The second question is more difficult, since relevant words in the question are not directly present in the text, e.g. 'pay extra' in the question relates to 'supplementary charge' in the text. This requires us to find a way to find similar words and phrases, i.e. calculate semantic similarity. Also, since we have an question that is not a full sentence (by the lack of the 'punct' type), we can compose it with each answer and check for truth.

From the dependency parse of the question, we can see that 'pay' is the relevant verb (via xcomp from root) and 'bathroom' as the object. Since the word bathroom does not appear in the text, we can use a semantic similarity measure to look for the most similar words.

Word count: 322 [1]

Task 2

Task 1 refers to semantic similarity measures, which need to be used when words occur in questions that don't occur in the text. e.g. 'drip'. I discuss two methods here, wordnet and word2vec, and show that they find 'droplet' is the most similar word to 'drip', and

[1] All wordcounts calculated using texcount

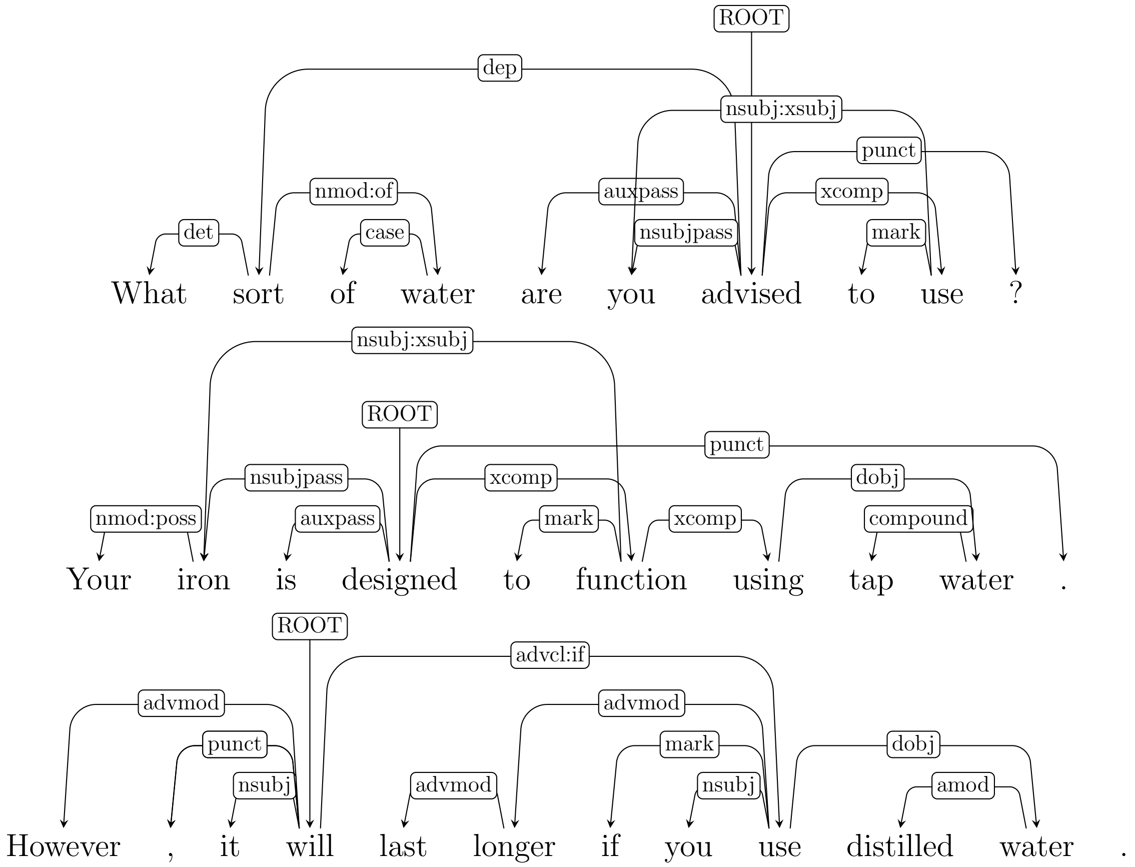

Figure 1: dependency parses relating to question 1 of text 1

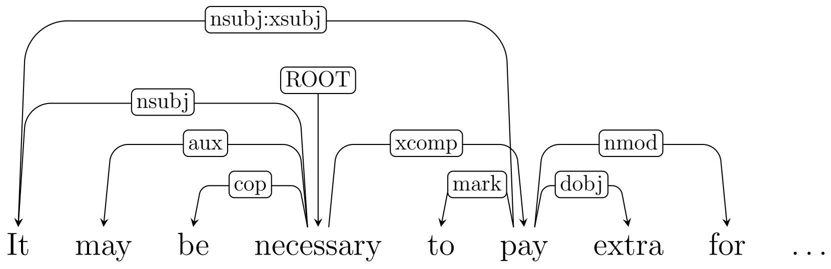

Figure 2: Dependency parse relating to question 12 of text 2

so the answer to the question lies in the sentence conatining the word ‘droplet’. Being able to derive an answer from this is more difficult, and requires chains of reasoning (task 3).

Wordnet is a large database of english words grouped into ‘synsets’, which represent distinct concepts. These synsets are linked by lexical relations[8]. The result is a large network from which related words can be obtained. To use it here, we must first carry out POS tagging [1] to disambiguate words. The word ‘drip’ as a verb belongs to 2 synsets, and by traversing derivationally related forms of the words in these synsets, we arrive at droplet in 2 steps, which is the closest word in the text. Length of chains is one metric for similarity, but other similarity metrics can be used (Liu et al. 2015) [4].

Word2Vec[3] is another method by which similar words can be found. Word2vec represents words as n-dimensional vectors which are not directly interpretable, using a neural model to learn these ‘word embeddings’ [9]. This method can also be extended to phrases in a meaningful way [5], enabling us to detect similarity between phrases. Using Word2Vec, we can convert each word found in the text to a word embedding and calculate the cosine similarity - carrying this out on the text shows that the most similar word is ’droplet’ (0.390).

Looking at question 12 from text 2, our system must be able to find similarity between the ‘pay extra’ and ‘supplementary charge’, as well as ‘private facilities’ and ‘bathroom’. This requires the use of collocations, supported by both word2vec and wordnet - however, neither the pre-trained word2vec embeddings nor wordnet contain these phrases as collocations. In the case of word2vec, more training may overcome this. However, both show similarities between ‘pay’ and ‘charge’.

Word count: 344

Task 3

To answer question 5, an example of an informal reasoning chain is:

- Removing creases implies using the iron
- An iron is hot when being used
- Clothes worn by a person are in direct touch with skin
- Using an iron on these clothes means the iron is in contact with skin
- Skin can burn if touched by a hot surface
- So the skin is at risk of being burned
- Burning causes hurt

This chain ends with the answer corresponding to the sentence *remove creases from an item of clothing that is being worn.*

This reasoning requires a large amount of knowledge which cannot be gained directly from the text, as well as interpreting parts of the question - e.g. ‘misuse’ should be similar to ‘do not attempt’. Some parts of the first two tasks could be used for this, for example, understanding that we are looking for the object of the verb misuse suggests we should look at the dependents of ’misuse’ (Task 1). Since the word is not directly present, we must use a semantic similarity measure (Task 2). The difficulty now comes from automating the reasoning procedure.

We could use compositional semantics and a logical representation for this. The steps above as well as the question could be represented in first order logic, with inference being performed to find a ‘proof’ of the query [7].

However, this would require a very large knowledge base, including general knowledge from outside the data given - for example, knowing that skin burns if touched by something hot is something that can’t be learned by just looking at the text in question.

While there are systems that use this technique for certain domains, such as logAnswer [2] or ASP [6], it would intractable to build a large enough knowledge base that would be able to answer general queries.

Overall, a rule-based system built to answer questions of this type and produce a reasoning chain would be possible, but is unlikely to be robust to different types of texts or questions.

Word count: 332

References

[1] Paula Buttery. Lecture 2: Morphology and finite state techniques, 2019.

[2] Ulrich Furbach, Ingo Glöckner, Hermann Helbig, and Björn Pelzer. Loganswer-a deduction-based question answering system (system description). In *International Joint Conference on Automated Reasoning*, pages 139–146. Springer, 2008.

[3] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. *CoRR*, abs/1405.4053, 2014.

[4] X. Liu, Y. Zhou, and R. Zheng. Measuring semantic similarity in wordnet. In *2007 International Conference on Machine Learning and Cybernetics*, volume 6, pages 3431–3435, Aug 2007.

[5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[6] Arindam Mitra, Peter Clark, Oyvind Tafjord, and Chitta Baral. Declarative question answering over knowledge bases containing natural language text with answer set programming. *CoRR*, abs/1905.00198, 2019.

[7] Simone Teufel, Ann Copestake, and Ryan Cotterell. Lecture 6: Compositional semantics, 2019.

[8] Simone Teufel, Ann Copestake, and Ryan Cotterell. Lecture 7: Lexical semantics, 2019.

[9] Simone Teufel, Ann Copestake, and Ryan Cotterell. Lecture 8: Distributional semantics, 2019.