Apples to Apples: Learning Semantics of Common Entities Through a Novel Comprehension Task

Omid Bakhshandeh
University of Rochester
omidb@cs.rochester.edu

James F. Allen
University of Rochester
Institute for Human and Machine Cognition
james@cs.rochester.edu

Abstract

Understanding common entities and their attributes is a primary requirement for any system that comprehends natural language. In order to enable learning about common entities, we introduce a novel machine comprehension task, GuessTwo: given a short paragraph comparing different aspects of two real-world semantically-similar entities, a system should guess what those entities are. Accomplishing this task requires deep language understanding which enables inference, connecting each comparison paragraph to different levels of knowledge about world entities and their attributes. So far we have crowdsourced a dataset of more than 14K comparison paragraphs comparing entities from a variety of categories such as fruits and animals. We have designed two schemes for evaluation: open-ended, and binary-choice prediction. For benchmarking further progress in the task, we have collected a set of paragraphs as the test set on which human can accomplish the task with an accuracy of 94.2% on open-ended prediction. We have implemented various models for tackling the task, ranging from semantic-driven to neural models. The semantic-driven approach outperforms the neural models, however, the results indicate that the task is very challenging across the models.

1 Introduction

In the past few years, there has been great progress on core NLP tasks (e.g., parsing and part of speech tagging) which has renewed interest in primary language learning tasks which require text understanding and reasoning, such as machine comprehension (Schoenick et al., 2016; Hermann et al., 2015; Rajpurkar et al., 2016; Mostafazadeh et al., 2016). Our question is *how far have we got in learning basic concepts of the world through language comprehension*. If we look at the large body of work on extracting knowledge from unstructured corpora, we will see that they often lack some very basic pieces of information. For example, let us focus on the basic concept of *apple*, the fruit. What do the state-of-the-art systems and resources know about an *apple*? None of the state-of-the-art knowledge bases (Speer and Havasi, 2012; Carlson et al., 2010; Fader et al., 2011) include much precise information about the fact that apples have an edible skin, vary from sweet to sour, are round, and relatively the same size of a fist. Moreover, there is no clear approach on how to extract such information, if any, from trained word embeddings. This paper focuses on how we can automatically learn about various attributes of such generic entities in the world.

A key observation motivating this work is that we can learn more detail about objects when they are compared to other similar objects. When we compare things we often contrast, that is, we count their similarities along with their dissimilarities. This results in covering the primary attributes and aspects of objects. As humans, we tend to recall and mention the difference between things (say *green skin* vs. *red skin* in apples) as opposed to absolute measures (say the existence of skin). Interestingly, there is evidence that human knowledge is structured by *semantic similarity* and the relations among objects are defined by their relative perceptual and conceptual properties, such as their form, function, behavior, and environment (Collins and Loftus, 1975; Tversky and Gati, 1978; Cree and Mcrae, 2003). Our idea is to leverage comparison as a way of naturally learning

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pages 906–916
Vancouver, Canada, July 30 - August 4, 2017. ©2017 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17-1084

about common world concepts and their specific attributes.

Comparison, where we name the similarities and differences between things, is a unique cognitive ability in humans[1] which requires memorizing facts, experiencing things and integration of concepts of the world (Hazlitt, 1933). It is clear that developing AI systems that are capable of comprehending comparison is crucial. In this paper, in order to enable learning through comparison, we introduce a new language comprehension task which requires understanding different attributes of basic entities that are being compared.

The contributions of this paper are as follows: (1) To equip learning about common entities through comparison comprehension, we have crowdsourced a dataset of more than 14K comparison paragraphs comparing entities from nine broad categories (Section 2). This resource will be expanded over time and will be released to the public. (2) We introduce a novel task called GuessTwo, in which given a short paragraph comparing two entities, a system should guess what the two things are. (Section 3). To make systematic benchmarking on the task possible, we vet a collection of comparison paragraphs to obtain a test set on which human performs with an accuracy 94.2%. (3) We present a host of neural approaches and a novel semantic-driven model for tackling the GuessTwo task (Sections 4, 5). Our experiments show that the semantic approach outperforms the neural models. The results strongly suggest that closing the gap between system and human performances requires richer semantic processing (Section 6). We hope that this work will establish a new base for a machine comprehension test that requires systems to go beyond information extraction and towards levels of performing basic reasoning.

2 Data Collection

To enable learning about common entities, we aimed to create a dataset which meets the following goals:

1. The dataset should be a collection of high-quality documents which are rich in comparing and contrasting entities using their various attributes and aspects.
2. The comparisons in the dataset should involve everyday non-technical concepts, making their comprehension easy and commonsense for a human.

After many experiments with scraping existing Web resources, we decided to crowdsource the comparison paragraphs using Amazon Mechanical Turk[2] (Mturk). We prompt the crowd workers as follows: "Your task is to compare two given items in one simple language paragraph so that a knowledgeable person who reads it can guess what the two things are". The workers were instructed to compare only the major and well-known aspects of the two entities. We also asked them to use $\mathbb{X}$ and $\mathbb{Y}$ for anonymously referring to the two entities. Table 1 shows three examples of our crowdsourced comparison paragraphs. As these examples show, the paragraphs are very contentful and rich in comparison which meets our initial goals in the dataset creation.

Entity Pair Selection. The choice of the two entities which should be compared against each other plays a key role in the quality of the collected dataset. It is evident that naturally, we compare two things which are semantically similar, yet have some dissimilarities[3], such as *jam* and *jelly*. Given the goals of our task, we experimented with concrete nouns which share a common taxonomy class. We choose semantic classes which have at least five well-known entities. So far, we have covered nine broad categories as shown in Figure 2, with 21 subcategories shown in Figure 3. We use Wikipedia item categories and the WordNet (Miller, 1995) ontology for identifying entities from each subcategory. Then, we choose the most common entities by looking up their frequency on Google Web 1T N-grams[4]. We manually inspected the frequency-filtered list to make sure that the entities are rather easy to describe without getting technical. Given the list of entities, we paired each entity with at most five and at least three other entities from the same subcategory. We also include inter-subcategory compar-

[1] It has been suggested (Hazlitt, 1933) that children under seven years old cannot name differences between simple things such as *peach* and *apple*. This further shows that the ability for comparison develops at a later age and is cognitively complex.

[2] www.mturk.com

[3] Tversky's (1978) analysis of similarity suggests that similarity statements compare objects that belong to the same class of things.

[4] https://catalog.ldc.upenn.edu/ldc2006t13

Comparison Paragraph	**Entity 𝕏**	**Entity 𝕐**
Both 𝕏 and 𝕐 are fruits and a variety of apples. 𝕏 and 𝕐 are generally similar in size. 𝕏 are dark red in color when ripe, while 𝕐 are a bright green color. 𝕏 is sweeter and softer than 𝕐 in taste and texture, sometimes starchy. 𝕐 are tart and somewhat stringy. 𝕐 is often used in cooking, whereas 𝕏 is not.	*Red Delicious* Apple Fruit	*Granny Smith* Apple Fruit
The 𝕏 and 𝕐 are two types of vehicles. 𝕏 is a smaller vehicle than 𝕐. The 𝕏 has two wheels while 𝕐 has none. The 𝕏 travels on roadways and smooth surfaces, whereas 𝕐 is capable of flying. Only one or two people are able to ride on 𝕏 at once, while 𝕐 can carry more people.	*Motorcycle* Motor Vehicle Vehicle	*Helicopter* Aircraft Vehicle
𝕏 and 𝕐 are both types of world cuisines. 𝕏 incorporates a lot of pasta dishes and sauces, with basil, tomato, and cheese being major ingredients. 𝕐 consists of many curries and stir fried dishes, with coconut and lemongrass being used often. 𝕐 is generally spicier and more aromatic than 𝕏. 𝕏 is a European cuisine, while 𝕐 is an Asian cuisine.	*Italian Cuisine* Cuisine Cuisine	*Thai Cuisine* Cuisine Cuisine

Table 1: Examples from the GuessTwo comprehension dataset. Also provided with the dataset is the subcategory and the broad category of the entities which are listed below the entity names in this Table.

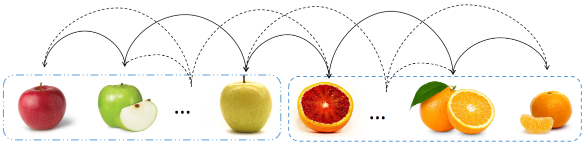


Figure 1: An example illustrating the entity pair matching process.

ison for a handful of entities at the boundaries. Figure 1 illustrates our entity pair matching process with an example on subcategories 'apple' and 'citrus'.

Data Quality Control. Our task of free-form writing is trickier than many other tasks such as tagging on Mturk. To instruct the non-expert workers, we designed a qualification test on Mturk in which the workers had to judge whether or not a given paragraph is acceptable according to our criteria. We used three carefully selected paragraphs to be a part of the qualification test. Moreover, to further ensure the quality of the submissions, one of our team members qualitatively browsed through the submissions and gave the workers detailed feedback before approving their paragraphs.

For each pair of entities, we collected eight comparison paragraphs from different workers. Given that different workers have different perspectives on what the major aspects to be compared are, collecting multiple paragraphs helps further enriching our dataset. We constrained the paragraphs to be at least 250 characters and at most 850 characters. Table 2 shows the basic statistics of our dataset. In this Table, we also included the median number of adjectives (including comparatives) per paragraph as a measure of descriptiveness of the comparison paragraphs. As a point of reference, the median number of adjectives in a random Wikipedia paragraph of the same length is 5.

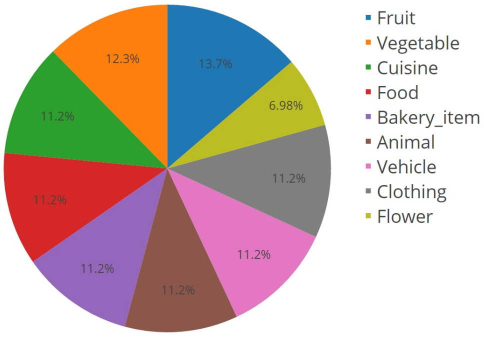


Figure 2: Distribution of broad category of the entities.

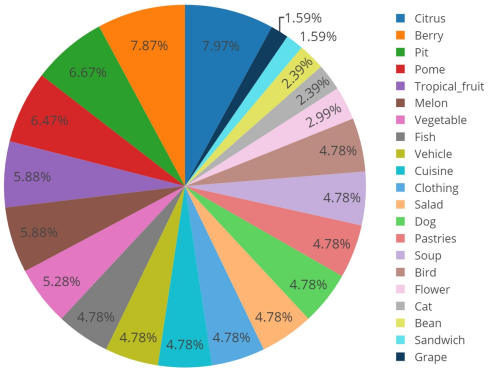


Figure 3: Distribution of subcategory of the entities.

Given the quality control we have in place, our data collection is going slowly. So far we have collected 14,142 paragraphs; however, we are aiming

Number of total approved paragraphs	**14,142**
Number of workers participated	649
Average number of paragraphs by one worker	21.7
Average work time among workers (minutes)	17.3
Median work time among workers (minutes)	6.4
Payment per paragraph (cents)	50
Number of broad entity categories	9
Number of entity sub-categories	24
Number of unique entities	920
Number of unique pairs compared	1974
Median number of sentences per paragraph	7
Median number of tokens per paragraph	70
Median number of adjectives per paragraph	7

Table 2: Statistics of the GuessTwo dataset as of April 2017.

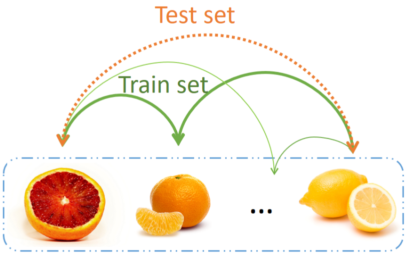


Figure 4: An example showing the entity pairs in the test and training sets.

to expand the resource over time.

Test Set Creation. In order to enable benchmarking on the task, we assessed the quality of a random sample of GuessTwo paragraphs as follows: we show the paragraph to three human workers on Mturk and ask them to guess what the two things are. Then, we choose 520 paragraphs for which all three workers have made exactly correct guesses for both entities. The test set will also be expanded along with the further data collection.

We divided the rest of the GuessTwo dataset into training and validation sets, with a 90%/10% split. To ensure that the test set requires some level of basic reasoning, our training set does not share any exact entity pairs with the validation or test set. This further enforces systems to learn about entities indirectly by processing across paragraphs. For instance, as shown in Figure 4, at test time, a system should be able to guess a comparison involving the entities *blood orange vs. lemon* by having seen comparisons of *blood orange vs. tangerine* and *tangerine vs. lemon*.

Our dataset will be released to the public through `https://omidb.github.io/guesstwo/`.

3 The GuessTwo Task Definition

We define the following two different schemes for the GuessTwo task:

- **Open-ended GuessTwo.** Given a short paragraph P which compares two entities $\mathbb{X}$ and $\mathbb{Y}$, guess what the two entities are. The scope of this prediction is the set of all entities appearing in the training dataset.
- **Binary Choice GuessTwo.** Given a short paragraph P which compares two entities $\mathbb{X}$ and $\mathbb{Y}$, and two nominals n_1 and n_2, choose 0 if $n_1 = \mathbb{X}$ and $n_2 = \mathbb{Y}$, choose 1 otherwise.

We speculate that system which can successfully tackle the GuessTwo task, has achieved two major objectives: (1) Has successfully learned the knowledge about entities stored in any form (e.g., continuous-space representation or symbolic) (2) Has a basic natural language understanding capability, using which, it can comprehend a paragraph and access its knowledge. We predict that our training dataset has enough detailed information about entities for learning the required knowledge for tackling the task. Given the design of our dataset, at test time, a system should perform some level of reasoning to go beyond understanding only one paragraph.

4 Neural Models

In this Section we present various end-to-end neural models for tackling the task of GuessTwo.

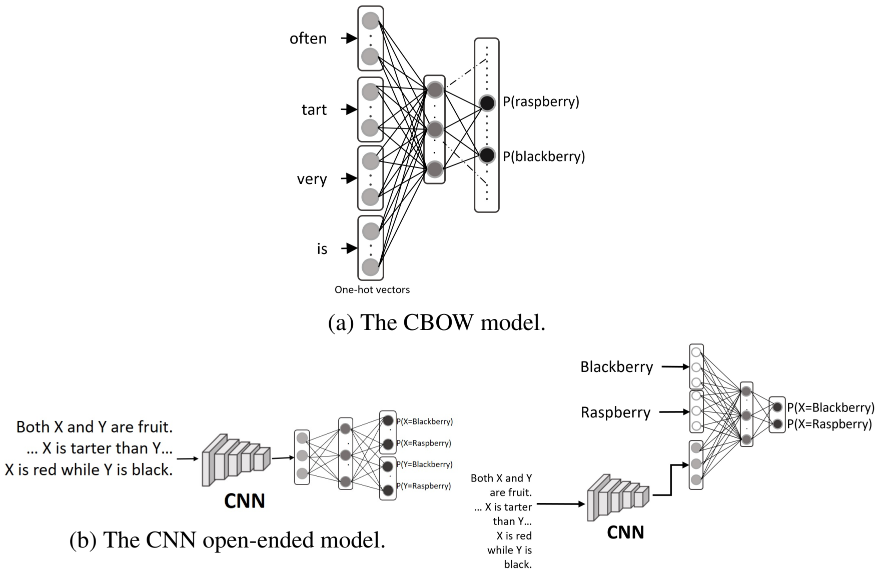
Continuous Bag-of-words Language Model. This model computes the probability of a sequence of consecutive words in context. The premise is that the probability of a paragraph with the correct realization of $\mathbb{X}$ and $\mathbb{Y}$ should be higher than the a paragraph with incorrect realizations. In order to compute the probability of a word given a context we use Continuous Bag-of-words (CBOW) (Mikolov et al., 2013a) which models the following conditional probability:

$$p(w|C(w), \theta) \qquad (1)$$

here, $C(w)$ is the context of the word w and θ is the model parameters. Then, the probability of a sequence of words (in a paragraph) is computed as follows:

$$\prod_{i=1}^{n} p(w_i|C(w_i), \theta) \qquad (2)$$

We define context to be a window of five words. Figure 5a summarizes this model. We train this



(a) The CBOW model.

(b) The CNN open-ended model.

(c) The CNN binary-choice model.

Figure 5: Various neural models for tackling the task of GuessTwo.

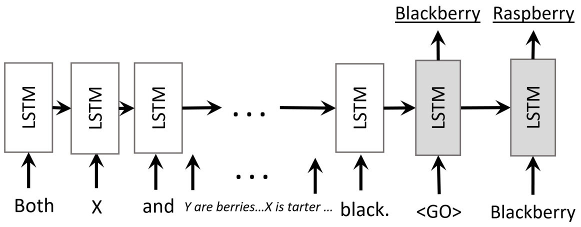


Figure 6: The Encoder-Decoder model.

model on two datasets: (1) A collection[5] of processed Wikipedia articles. Wikipedia articles often include definitions and descriptions of variety of items, which can provide a reasonable resource for our task. (2) the GuessTwo training dataset. We call these models *CBOW-Wikipedia* and *CBOW-GuessTwo* respectively.

At test time, for open-ended prediction we find the two nominals which maximize the following probability:

$$\underset{x,y}{\operatorname{argmax}} \prod_{i=1}^{n} p(w_i | C(w_i)_{x,y}, \theta) \qquad (3)$$

where $C(w_i)_{x,y}$ indicates the context in which any occurrences of $\mathbb{X}$ have been replaced with x and $\mathbb{Y}$'s have been replaced with y. For binary choice classification, we use the same modeling except that we only consider $x = n_1, y = n_2$ and $x = n_2, y = n_1$.

Encoder-Decoder Recurrent Neural Net (RNN). This model is a sequence-to-sequence generation model (Cho et al., 2014; Sutskever et al., 2014) that maps an input sequence to an output sequence using an encoder-decoder RNN with attention (Bahdanau et al., 2014). The encoder RNN processes the comparison paragraph and the decoder generates the first item followed by the second item (Figure 6). The paragraph is encoded into a state vector of size 512. This vector is then set as the initial recurrent state of the decoder. We tune the model parameters on the validation set, where we set the number of layers to 2. The model is trained end-to-end, using Stochastic Gradient Descent with early stopping.

For open-ended prediction, we use beam search with beam-width = 25 and then output the two tokens with the highest probability. For binary choice classification, we use the same model where we set the encoder RNN inputs to the input paragraph tokens, then, we set the input of the decoder RNN once to [n_1, n_2] and next to [n_2, n_1]. After running the network forward, we take the probability of the decoder logits and choose the ordering which has the highest probability.

Convolutional Neural Network (CNN) Encoder. As shown in the Figure 5b, this model first uses a Convolutional Neural Network (CNN) (LeCun and Bengio, 1998) for encoding the paragraph (Kim, 2014). We train a simple CNN with one layer of convolution on top of pre-trained word vectors. Here we use the word vectors trained by

[5] http://mattmahoney.net/dc/text8.zip

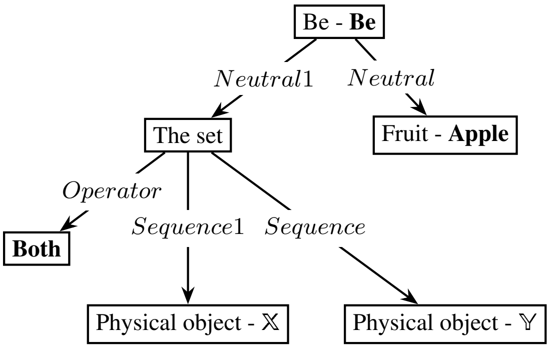


Figure 7: Semantic parsing for the sentence *Both $\mathbb{X}$ and $\mathbb{Y}$ are apples.*

Skip-gram model (Mikolov et al., 2013b) on 100 billion words of Google News[6]. For open-ended prediction, the output of CNN is fed forward and transformed into a 300 dimension vector. Then, we use a softmax layer to get the probability of each of the possible nominals for $\mathbb{X}$ and $\mathbb{Y}$. For binary choice classification, we use the same architecture and settings as above. Additionally, we encode each nominal into a 300-dimensional vector, which then gets concatenated with the paragraph vector. Figure 5c shows this model.

5 Semantic-driven Model

In this Section we present a semantic-driven approach which models the comparison paragraph using semantic features and is capable of performing basic reasoning across paragraphs.

5.1 Representing Paragraphs

The question is, given a comparison paragraph, what is the best representation which can enable further reasoning? The comparison paragraphs often have complex syntactic and semantic structures, which might be challenging for many off-the-shelf NLP tools to process. For instance, consider the sentence $\mathbb{X}$ *is much sweeter in taste than* $\mathbb{Y}$. Although a dependency parser provides a lot of information regarding how the individual words relate grammatically, it does not give us any information regarding how $\mathbb{Y}$'s sweetness (which is elided from the sentence and is implicit) relates to $\mathbb{X}$'s. As another processing technique, if we use the standard information extraction methods for extracting and representing syntactic triplets (*argument1, relation, argument2*) (Fader et al., 2014; Etzioni et al., 2011), we will extract a triplet such as $\mathbb{X}$ *is sweeter* which shares the same shortcomings.

Our approach for better representation of comparison paragraphs starts with a broad-coverage semantic parser (Banarescu et al., 2013; Bos, 2008; Allen et al., 2008). A semantic parser maps an input sentence to its formal meaning representation, operating at the generic natural language level. Here we use the TRIPS[7] (Allen et al., 2008) broad-coverage semantic parser. TRIPS provides a very rich semantic structure; mainly it provides sense disambiguated deep structures augmented with semantic ontology types. Figure 7 shows an example TRIPS semantic parse. In this graph representation, each node specifies a word in bold along with its corresponding ontology type on its left. The edges in the graph are semantic roles[8]. As you can see, this semantic parse represents the sentence by decoupling the token 'both' and attributing the property of 'be apple' to both $\mathbb{X}$ and $\mathbb{Y}$.

In our comparison paragraphs there are two major types of sentences:

- **Sentences with Absolute Information.** These sentences contain direct information about the entities, such as $\mathbb{X}$ *is red* or *Both* $\mathbb{X}$ *and* $\mathbb{Y}$ *are very sweet*. From each absolute sentence, we extract frames which describe the absolute attributes of the corresponding entity. We define a frame to be a subgraph of a semantic parse which involves exactly one entity and all of its semantic roles. Relying on the deep semantic features offered by the semantic parser, we perform negation propagation[9] and sequence decoupling, among others features. For example, given a sentence which has a sequence, as the one depicted in Figure 7, we perform sequence decoupling and extract the two frames *[$\mathbb{X}$ Be Apple]* and *[$\mathbb{Y}$ Be Apple]*.
- **Sentences with Relative Information.** These sentences contain relative information about the two entities, for instance, $\mathbb{X}$ *is somewhat sweeter than* $\mathbb{Y}$. As opposed to the sentences with absolute information, we cannot extract frames from sentences with comparisons directly. Various properties of entities can be associated with an abstract scale, such as 'size' or 'sweetness', on which dif-

[6] https://code.google.com/archive/p/word2vec/

[7] http://trips.ihmc.us/parser/cgi/parse

[8] Refer to http://trips.ihmc.us/parser/LFDocumentation.pdf for the full list of semantic roles in TRIPS parser.

[9] A common construction which needs negation propagation is *Neither* $\mathbb{X}$ *nor* $\mathbb{Y}$ *are ...* .

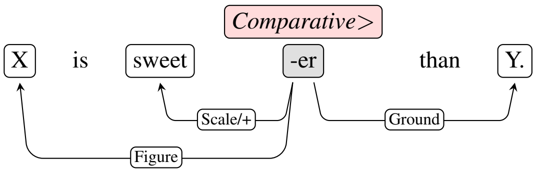


Figure 8: The comparison construction predicted for the sentence *X is sweeter than Y.*

ferent entities can be compared. In order to extract such scales and the relative standing of items on them we use the structured prediction model presented in Bakhshandeh et al. (2016), which given a sentence predicts its comparison structures. Figure 8 shows an example predicate-argument structure that is predicted by this model. We use pretrained model on the annotated corpus (Bakhshandeh et al., 2016) of comparison structures.

Given a comparison structure such as the one presented in Figure 8, we can extract the information that on the scale of 'sweetness' $\mathbb{X}$ is higher than $\mathbb{Y}$. It is clear that one can build a large knowledge base of such relations by reading large collections of comparison paragraphs. We populate our knowledge base of relative information about entities as follows: First, we predict the comparison structure of each sentence and then extract a binary relation $\prec_s$ which shows the relation on the scale of s. Second, for any scale s, we apply transitivity on its entities. As shown in equation 4, the binary relation $\prec_s$ is transitive over the set of all entities, A. This process, called closure, enables us do basic reasoning and derives implicit relations on scales from explicit relations.

$$\forall s \in S \; \forall x, y, z \in A : (x \prec_s y \; \wedge \; y \prec_s z) \\ \implies x \prec_s z \tag{4}$$

The product of this step is a structured knowledge base on entity ordering which we call the *ordering lattice*. Figure 9 shows an example partial ordering lattice inferred by our model, where the sweetness of *Golden Delicious* can be compared to *Granny Smith* through their direct link with *Red Delicious*.

5.2 Modeling

Given a paragraph P, we first extract the set of all the absolute information frames for $\mathbb{X}$ and $\mathbb{Y}$ (as described above), called $F_{\mathbb{X}}(P)$ and $F_{\mathbb{Y}}(P)$. Second, for the sentences with relative information, we extract all the binary relations $\prec_s \in R(P)$ that should hold between $\mathbb{X}$ and $\mathbb{Y}$. Then, our objective is to find two realizations for $\mathbb{X}$ and $\mathbb{Y}$ that maximize the following:

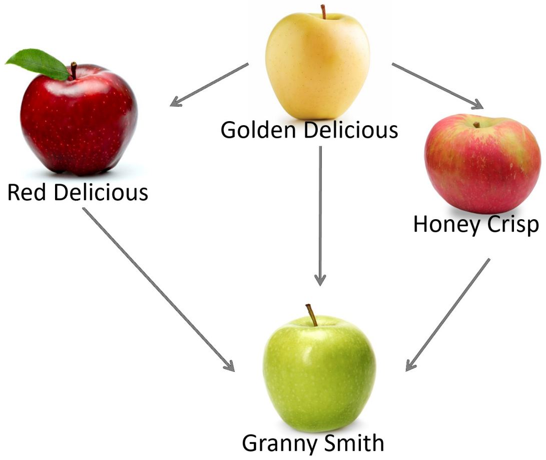


Figure 9: The inferred partial ordering lattice comparing the *sweetness* of different apples.

$$\underset{x,y}{\text{argmax}} \; p(x|F_{\mathbb{X}}(P)) + p(y|F_{\mathbb{Y}}(P)) \\ \text{s.t. } \forall \prec_s \in R(P) : x \prec_s y \tag{5}$$

In order to compute the $p(x|F_{\mathbb{X}}(P))$ and $p(y|F_{\mathbb{Y}}(P))$ scores we used Regularized Gradient Boosting (XGBoost) classifier (Friedman, 2000), which uses a regularized model formulation to limit overfitting. We directly use each frame in the $F_{\mathbb{X}}(P)$ and $F_{\mathbb{Y}}(P)$ sets as the classifier features. We use Integer Linear Programming (ILP) for formulating the constraints as follows: for each relation $r \in R$ on the scale s, we lookup the scale s in the ordering lattice and make the blacklist $B(P)$ containing each pair of entities which do not satisfy the relation r. Our ordering lattice does not have perfect complete information, hence, we have Open World Assumption and only prune our search space not to include the already observed pairs which violate the relation. our ILP objective function will be the following:

$$\underset{b,b'}{\text{argmax}} \sum_{x \in N} b_x \, p(x|F_{\mathbb{X}}(P)) + \\ \sum_{y \in N} b'_y \, p(y|F_{\mathbb{Y}}(P)) \\ \text{s.t. } \forall \, (j, j') \in B(P) : b_j + b'_{j'} \leq 1 \tag{6}$$

where N is the set of all possible realizations and b and b' are the binary indicator variables, so $b_x = 1$ indicates the realization of x for $\mathbb{X}$.

In the case of open-ended prediction, the maximization presented in Equation 6 is carried out on the set N. In the case of binary choice classification, however, only the two choices of n_1 and n_2 are considered in the maximization.

6 Results

We evaluate all the models presented in Sections 4 and 5 using the following accuracy measure:

$$\frac{\#\text{correct predictions of both entities}}{\#\text{test cases}} \tag{7}$$

As for the open-ended prediction we compute the nominator of the accuracy measure using three various matching methods on both entities: (1) exact-match, (2) subcategory match, (3) broad category match.

As Table 3 shows, the semantic model outperforms all the neural models. Moreover, the ILP constraints have been very effective in directing the system in the correct search space. Among the neural models, the Encoder-Decoder RNN model performs noticeably better than other models when matching the subcategory and broad category. According to the exact-matching, neither of the CBOW models could guess any of the two test entities correctly. Overall, it is evident that the end-to-end neural models have not been able to generalize well and learn about the attributes of entities across various training paragraphs. This can be partly due to not being trained on large enough comparison training dataset. The semantic model, however, could outperform the neural models using the same amount of data. To a degree, this is because the semantic model leverages the basic language understanding capabilities offered by the semantic parser.

It is also important to note that our semantic approach is not only capable of binary and open-ended prediction, but it also offers two byproducts that can be used as knowledge in a variety of other tasks: (1) a set of the most important absolute information frames which can be chosen based on feature importance in the classification, (2) the partial ordering lattice of entities. Overall, the results strongly suggest that the GuessTwo task is challenging, with the open-ended scheme being the most challenging. There is a wide gap between human and system performance on this task, which makes it a very promising task for the community to pursue.

Model	**Binary**	**Open-ended**	
		Exact.	**Subcat.**
Human	100.0	94.2	100.0
CBOW-Wikipedia	51.9	0.0	1.5
CBOW-GuessTwo	51.7	0.0	1.1
Encoder-Decoder RNN	58.8	2.9	6.8
CNN	57.6	1.9	2.5
Semantic (no constraints)	61.5	10.5	38.5
Semantic (with ILP constraints)	**69.2**	**11.7**	**40.4**

Table 3: System accuracy results on the GuessTwo test set. A random baseline on binary choice task achieves 51%. The open-ended evaluation has two columns: exact-match (exact) and subcategory match (subcat), respectively.

7 Related Work

The task of Machine Comprehension (MC) has gained a significant attention over the past few years. The major driver for MC has been the publicly available benchmarking datasets. A variety of MC tasks have been introduced in the community (Richardson et al.; Hermann et al., 2015; Rajpurkar et al., 2016; Hill et al., 2015), in which the system reads a short text and answers a few multiple-choice questions. The reading comprehension involved in these tests ranges from reading a short fictional story (Richardson et al.) to reading a short news article (Hermann et al., 2015). In comparison, in the GuessTwo task the reading comprehension involves reading a short comparison paragraph and one can say the multiple-choice question is the constant *What are* $\mathbb{X}$ *and* $\mathbb{Y}$*?*

The CNN/DailyMail dataset consists of more than 100K short news articles with the questions automatically created from the bullet-point summaries of the original article. This dataset uses fill-in-the-blank-style questions such as 'Producer *X* will not press charges against Jeremy Clarkson' where the system should choose among all the anonymized entities in the corresponding paragraph to fill in *X*. The Stanford Question Answering (SQuAD) dataset is another recent machine comprehension test with over 500 Wikipedia articles and +100,000 crowdsourced questions. The answer to every question in this dataset is a span of text from the corresponding reading passage.

Human accuracy on CNN/DailyMail is estimated to be around 75% (Chen et al., 2016) with the current state-of-the-art at 76.1 on CNN (Sordoni et al., 2016), and 75.8 on DailyMail (Chen et al., 2016). The human F1 score on SQuAD dataset is reported to be at 86.8%, with the current state-of-the-art achieving 82.9%. Given these statistics, neither of these datasets leave enough room for further research. Given that in both these tasks the answer to the question is directly found in the provided passage, we argue that the community requires a more challenging MC task which goes beyond matching and needs some level of inference across passages. The GuessTwo task requires basic reasoning and inference across paragraphs for comprehending various aspects of entities relative to one another.

Another interesting task is MCTest (Richardson et al.), which is a reading comprehension test with 660 fictional stories as the passage and four questions per story. The human-level performance on MCTest is estimated to be around 90%, with the state-of-the-art achieving an accuracy of 70% (Wang et al., 2015). MCTest is also proven to be challenging, however, given its very limited training data, further progress on the task has been hindered. Yet another relevant QA task is the Allen AI Science Challenge (Clarke et al., 2010; Schoenick et al., 2016), which is a dataset of multiple-choice questions and answers from a standardized 8th grade science exam. The questions can range from simple fact lookup to complex ones which require extensive world knowledge and commonsense reasoning. This task requires machine reading of a variety of resources such as textbooks and goes beyond reading a couple of passages.

8 Conclusion

We introduced the novel task of GuessTwo, in which given a short paragraph comparing two common entities, a system should guess what the two entities are. The comparison paragraphs often have complex semantic structures which make this comprehension task demanding. Furthermore, guessing the two entities requires a system to go beyond only understanding one given passage and requires reasoning across paragraphs, which is one of the most under-explored, yet crucial, capabilities of an intelligent agent.

So far, we have crowdsourced a dataset of more than 14K comparison paragraphs comparing entities from nine major categories. For benchmarking the progress, we filter a collection of these paragraphs to create a test set, on which humans perform with an accuracy of 94.2%. For continuing our data collection, we would like to have a targeted entity pair selection where we particularly collect the missing relations in our partial ordering lattice. We believe that this process can help developing more effective systems. For the most recent statistics of the dataset and the best performing systems please check this website.

We presented a host of neural models and a novel semantic-driven approach for tackling the task of GuessTwo. Our experiments show that the semantic approach outperforms the neural models by a large margin. The poor performance of the neural models we experimented with can motivate designing new architectures which are capable of performing basic reasoning across paragraphs. The results strongly suggest that bridging the gap between system and human performance on this task requires models with richer language representation and reasoning capabilities. As a future work, we would like to explore the feasibility of marrying our semantic and neural models to exploit the benefits that each of them has to offer.

9 Acknowledgments

This work was supported in part by Grant W911NF-15-1-0542 with the US Defense Advanced Research Projects Agency (DARPA) and the Army Research Office (ARO). We would like to thank Linxiuzhi Yang for her help in the data collection and anonymous reviewers for their insightful comments on this work. We specially thank William de Beaumont for his invaluable feedback on this paper. We also thank the inputs from Steven Piantadosi, Brad Mahon, and Gregory Carlson on cognitive aspects of comparison.

References

James F. Allen, Mary Swift, and Will de Beaumont. 2008. Deep semantic analysis of text. In *Proceedings of the 2008 Conference on Semantics in Text Processing*. Association for Computational Linguistics, Stroudsburg, PA, USA, STEP '08, pages 343–354. http://dl.acm.org/citation.cfm?id=1626481.1626508.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR* abs/1409.0473. http://arxiv.org/abs/1409.0473.

Omid Bakhshandeh, Alexis Cornelia Wellwood, and James Allen. 2016. Learning to jointly predict ellipsis and comparison structures. In *Proceedings of The 20th SIGNLL Conference on Computational*

Natural Language Learning. Association for Computational Linguistics, Berlin, Germany, pages 62–74. http://www.aclweb.org/anthology/K16-1007.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*. Association for Computational Linguistics, Sofia, Bulgaria, pages 178–186. http://www.aclweb.org/anthology/W13-2322.

Johan Bos. 2008. Wide-coverage semantic analysis with boxer. In Johan Bos and Rodolfo Delmonte, editors, *Semantics in Text Processing. STEP 2008 Conference Proceedings*. College Publications, Research in Computational Semantics, pages 277–286.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, and Tom M. Mitchell. 2010. Toward an architecture for never-ending language learning. In *In AAAI*.

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the cnn/-daily mail reading comprehension task. In *Association for Computational Linguistics (ACL)*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* .

James Clarke, Dan Goldwasser, Ming-Wei Chang, and Dan Roth. 2010. Driving semantic parsing from the world's response. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, Stroudsburg, PA, USA, CoNLL '10, pages 18–27. http://dl.acm.org/citation.cfm?id=1870568.1870571.

Allan M. Collins and Elizabeth F. Loftus. 1975. A spreading-activation theory of semantic processing. *Psychological Review* 82(6):407 – 428.

George S. Cree and Ken Mcrae. 2003. Analyzing the factors underlying the structure and computation of the meaning of chipmunk, cherry, chisel, cheese, and cello (and many other such concrete nouns). *Journal of Experimental Psychology: General* 132(2):163–201+.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam Mausam. 2011. Open information extraction: The second generation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume One*. AAAI Press, IJCAI'11, pages 3–10. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-012.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Stroudsburg, PA, USA, EMNLP '11, pages 1535–1545. http://dl.acm.org/citation.cfm?id=2145432.2145596.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, KDD '14, pages 1156–1165. https://doi.org/10.1145/2623330.2623677.

Jerome H. Friedman. 2000. Greedy function approximation: A gradient boosting machine. *Annals of Statistics* 29:1189–1232.

V. Hazlitt. 1933. *The psychology of infancy*. E.P. Dutton and company, inc. https://books.google.com/books?id=I8svAAAAYAAJ.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*. pages 1693–1701.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *International Conference on Learning Representations (ICLR)* .

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, pages 1746–1751. http://aclweb.org/anthology/D/D14/D14-1181.pdf.

Yann LeCun and Yoshua Bengio. 1998. The handbook of brain theory and neural networks. MIT Press, Cambridge, MA, USA, chapter Convolutional Networks for Images, Speech, and Time Series, pages 255–258. http://dl.acm.org/citation.cfm?id=303568.303704.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781. http://arxiv.org/abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information*

Processing Systems 26, Curran Associates, Inc., pages 3111–3119. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM* 38(11):39–41. https://doi.org/10.1145/219717.219748.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, California, pages 839–849. http://www.aclweb.org/anthology/N16-1098.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text .

Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. ???? Mctest: A challenge dataset for the open-domain machine comprehension of text. pages 193–203.

Carissa Schoenick, Peter Clark, Oyvind Tafjord, Peter D. Turney, and Oren Etzioni. 2016. Moving beyond the turing test with the allen AI science challenge. *CoRR* abs/1604.04315. http://arxiv.org/abs/1604.04315.

Alessandro Sordoni, Phillip Bachman, and Yoshua Bengio. 2016. Iterative alternating neural attention for machine reading. *CoRR* abs/1606.02245. http://arxiv.org/abs/1606.02245.

Robert Speer and Catherine Havasi. 2012. Representing general relational knowledge in conceptnet 5. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uur Doan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), Istanbul, Turkey.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. pages 3104–3112. http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.

Amos Tversky and Itamar Gati. 1978. Studies of similarity. *Cognition and categorization* 1(1978):79–98.

Hai Wang, Mohit Bansal, Kevin Gimpel, and David A. McAllester. 2015. Machine comprehension with syntax, frames, and semantics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*. The Association for Computer Linguistics, pages 700–706. http://aclweb.org/anthology/P/P15/P15-2115.pdf.