Compositional semantics is the building up of phrasal or sentence meaning from the meaning of individual words by semantic rules, such as logical compositional semantics and distributional compositional semantics. These phrasal or sentence meaning can be used to perform various NLP tasks, like question answering, sentiment analysis, recognizing entailment, and summarization.

For a question answering task, we not only want a system to answer simple questions (e.g., "Where is New York City?") , but we also want it to answer complex questions (e.g., "What is the name of Brad Pitt’s ex wife?"). In a sentiment analysis task, we want to find meaning of the words and their combined meaning to determine whether the text is positive or negative. In a recognizing textual entailment task, the phrasal meaning is used to determine whether the text entails, contradicts, or neither entails nor contradicts the hypothesis.

1 Model Theoretic Semantics

Model theoretic semantics provides a way to link a sentence to an external representation of the world.

1.1 Description of components

There are three components to the model theoretic semantics: domain, properties, and relations. Domain are made up of entities. Properties defines and maps to these set of entities. Relations maps pair of entities. The combination of the components can be treated as database queries that allow users to ask specific questions about the world.
Logical forms are used to interpret the contents of the model. For example, "Mario’s is a Restaurant" can be written as $\text{restaurant}(\text{marios})$. $\lambda$-calculus is a variant of high-order logic, which treats arguments as functions. It is used to transform a sentence into logical form. The $\lambda$-calculus uses primitive types like Truth-value ($t$) and Entity ($e$).

Logical forms can also express conjoined properties. "Artusi and Majiko are expensive restaurants" can be written as $\text{expensive}(\text{artusi}) \land \text{restaurant}(\text{artusi}) \land \text{expensive}(\text{majiko}) \land \text{restaurant}(\text{majiko})$.

A binary relations of "Mario’s and Artusi serve italian food" can be expressed as $\text{cuisine}(\text{marios}, \text{italian}) \land \text{cuisine}(\text{artusi}, \text{italian})$.

First order predicate logic (FOPL) of "Slav likes expensive restaurants" can be expressed as $\forall x. \text{expensive}(x) \land \text{restaurant}(x) \Rightarrow \text{likes}(\text{slav}, x)$.

With logical form representation for questions, an evaluation for truth-value type can be answered. For example, the question "Is Artusui vegetarian" can be represented as $\text{vegetarian}(\text{artusui})$, which returns a truth-value.

Moreover, an example for entity to truth-value evaluation for the question "What restaurant do Slav, Emily, and David all like?" The question can be represented as $\lambda x. \text{likes}(\text{slav}, x) \land \text{likes}(\text{emily}, x) \land \text{likes}(\text{david}, x)$.

### 1.2 Semantic Parsers

Semantic parsers maps sentences onto logical forms to be used to interact with an external world. The words can be mapped to objects, properties, relations, logical operators, and first order predicate logic.

Show me flights from New York to London on a Monday morning.

$\lambda x \exists y. \text{flight}(x) \land \text{from}(x, \text{NY}) \land \text{to}(x, \text{LON}) \land \text{day}(x, \text{MON}) \land \text{time}(x, y) \land y < 12.00$

The usage of $\lambda$-calculus provides a clean way of combining semantic components.
A syntactic tree representation shows us how to order function application.

Combinatory Categorical Grammar is used to construct logical forms from a sentence. The leaf nodes of the CCG parse tree represents a pairing of a word and categories that represent its meaning. The categories include both syntactic (N, S, NP, ADJ, and PP) and semantic ($\lambda$-calculus expression) components of a word. The parent nodes represent combined meaning of their leaf nodes. At the end, the root node represents the combined meaning of the sentence. Other options for syntactic parses and semantic analyses are Dependency compositional semantics (DCS) and Synchronous context free grammars (SCFGs).

In order to score the derivations, it requires several features of words, such as logical form mapping, and grammar rules used. An option is to hire someone to label and map each logical form of a sentence.

Another method is to use an algorithm that takes input of Question/Answer pairs, lexicon, grammar (set of rules of any language of the world), knowledge base K (database of facts about the world), and randomized initial parameters. In the training phrase, the algorithm uses a factored difference between max scoring derivations and margin violating incorrect derivations to calculate cost.
2 Compositional Distributional Semantics

Distributional semantics use the co-occurrence statistics of words in a corpus to study the meaning of words and phrases in a language. Compositional distributional semantics is an extension to distributional semantic models that build longer unit of text out of elementary distributional representation. It uses vectors to represent the meanings of sentences.

2.1 Phrase Similarity Task

Phrase similarity task is used to give a number to represent how similar two words are with each other. The pre-trained word vectors can be combined to create phrase vectors through addition, multiplication, etc.

2.2 Bi-LSTM Model

The bi-LSTM model captures the combined weights from left-to-right and right-to-left order of a sentence for classification. This model ignores the traditional sentence of language modeling.
2.3 Entailment and Question Answering Tasks

In an entailment task, the goal is to use entailment, contradiction, or neutral relationship between a premise and a hypothesis to train a model. This is similar to the question and answering task that takes a hypothesis (or question) and premise (or document) and outputs a prediction.