UNIVERSITY OF LORRAINE

# SEMANTIC MINING
*defining the compatibility between the words*

Project Report
Methods for NLP

*Elena Khasanova, Esteban Marquer, Kelvin Han*
students of M1 NLP

January 2019

## Contents

# 1. Introduction

The word meaning representation is one of the most acute and complex tasks in natural language processing due to the unresolved aspects in semantic theory (what is meaning?) and the complexity of its representation with computational methods. Existing methods, with the word embeddings being the most popular, deal mostly with the relationships between the words and not the semantic "content". In this project, we attempt to develop an alternative methodology of meaning representation based on lexical and compositional semantics, which is predicted to be useful for a wide variety of NLP tasks, with sentence generation, disambiguation of synonyms and polysemous words, ontology development, translation, and many other. First, the theoretical basis for the system was investigated. Second, necessary computational algorithms and tools for deriving the meaning from the knowledge base (explanatory dictionary) were developed. Finally, the methods to evaluate the results obtained from the system were suggested.

# 2. Theoretical basis

The most common methods to deal with the meaning are distributional and compositional semantics. Distributional semantics has become the basis for the word embeddings, an approach based on the co-occurrence of lexical entities in the context, as is put by Firth[1], "a word is characterized by the company it keeps". Although this methodology shows impressive results in text analysis and text generation tasks, it can only predict some semantic properties, but embeddings do not contain any semantic information as it is. In the recent Twitter mega thread[2] on the meaning representation in NLP, researchers unanimously concluded that the currently exploited models learn not the meaning but the similarity in meaning. Professor of linguistics Emily Bender suggested that "If all the learner gets is text the learner cannot learn meaning".
In turn, the true meaning representation should (1) be linked to the external knowledge base, state of the world; (2) support computational inference, so that the combined concepts would allow deriving additional knowledge; (3) be unambiguous; (4) be expressive.

The model is thus required to learn the *grounded meaning* (learn the meaning of each expression as a state of the world state) and *lexical meaning* (e.g. learn how the meaning of sub-expressions compose together, as in logical forms)[3].

In this project, we attempt to develop *a methodology* for meaning representation that would comprise both the grounded and lexical meaning. Our approach builds on compositional and lexical semantics. The grounded meaning is represented with *semantic primes, which are defined as* a set of semantic concepts that are innately understood but cannot be expressed in simpler

[1] Firth, J.R. (1957). "A synopsis of linguistic theory 1930-1955". Studies in Linguistic Analysis: 1–32. Reprinted in F.R. Palmer, ed. (1968). Selected Papers of J.R. Firth 1952-1959. London: Longman.

[2] Leaning meaning in natural language processing: Twitter megathread. https://medium.com/huggingface/learning-meaning-in-natural-language-processing-the-semantics-mega-thread-9c0332dfe28e?fbclid=IwAR29kCGUo08phytwf5OfTwLr5hXod1fi_gKThDzFRUYBV1ILt9h91alcTKQ

[3] ibid.

terms, they represent words or phrases that are learned through practice but cannot be defined concretely. The lexical meaning is built out of the minimal units of meaning, or semes, according to the principles of compositionality. The database that contains the relationships between the lexical entities – words of the language – is represented as a monolingual explanatory dictionary. The potential of a dictionary for NLP tasks is explored in Gaume et al. (2006[4], 2004[5]). The semantic structure of a lexeme is revealed through iterative componential analysis and the method of step-by-step identification of the meaning[6]. According to this methodology, the minimal semantic components can be "mined" from the dictionary definitions: each meaningful word in the definition becomes a semantic component, then the definition is decomposed for each word until either a prime or a previously extracted component is met. The "meaningful" elements are those represented by nouns, adjectives, adverbs and verbs. The semes are organized hierarchically according to the frequency of occurrence in the definitions and possibly the depth in the mining process, this aspect needs further analysis. For instance, this is how the process of semantic mining would look for the word "question" (considering only the primary meaning):

*Question*

1. <u>something</u> that <u>someone</u> asks you when they <u>want</u> information

Components "something", "someone" and "want" are primes, so they are stored in the semantic structure, words "you", "when", "they" are not considered meaningful components in this case; words "ask" and "information" need further mining for meaningful components.

<u>Level 1:</u>

Ask

1. to **speak** or **write** to someone in order to **get** information from them

We see the components "someone" and information again, so their rank is increased. The problematic word here is "order" that would be considered by an automatic system a noun and thus will be included into later search, although normally prepositions are not taken into account. Words selected for definition decomposition are highlighted, primes are underlined.

Information

1. **knowledge** or **facts** about <u>someone</u> or <u>something</u>

Level 2

Speak

to **talk** to <u>someone</u> about <u>something</u>
Write

to **use** a **pen** to **make** <u>words</u>, **numbers**, or **symbols**
Get

to **obtain**, **receive**, or be **given** <u>something</u>

[4] Gaume, B. Venant, F., Victorri, B. Hierarchy in lexical organisation of natural languages. Denise Pumain. Hierarchy in Natural and social Sciences, 3, Springer, pp.121-142, 2006, Methodos Series.

[5] Gaume, B. Hathout, N., Muller, P. 2004. Word Sense Disambiguation using a dictionary for sense similarity measure. DOI: 10.3115/1220355.1220528. Accessed at: https://www.researchgate.net/publication/228601536

[6] The method is proposed by E. Kuznetsova in Lexicologia russkogo yazyka [The lexicology of the Russian language], 1982.

Knowledge

all the **facts** that <u>someone</u> <u>knows</u> about a **particular subject**

Facts

a **piece** of <u>true</u> **information**

Even at this level we see that words related to “information” - speaking, words, and interaction between “someone” and “something” are repeated multiple times and thus considered a core of the word semantics. If we dig further, there will appear components related to “achievement” or even “doing something with hands”, but they will be less frequent and consequently less related to the meaning, which seems intuitively correct. Performing the mining task manually, we can observe that, depending on the quality of the definitions and the complexity of the word semantics (e.g. it will be higher for abstract nouns and lower for concrete), the depth of the mining process can reach tens or even hundred levels, so, on the one hand, it is evident that the process would benefit from automation, and, on the other hand, an extra constraint on the number of levels can be introduced.

Our **goal** is to propose an alternative meaning representation model that will account for both the “grounded” and “lexical” meaning, to develop an automatic tool to extract semantic features from dictionary definitions and to build the hierarchical semantic structure of the word. Then, we will estimate the distance between the semantic trees. Further, we will use the semantic distance between the words as a measure of compatibility between the words to identify incompatible word pairings and generate non-contradictory word combinations. Our **solution** comprises the following steps: parsing online dictionary definitions, organizing semantic features into a graph and a tree structure, measuring distance between the trees, obtaining compatibility matrices, and finally, generating non-contradictory adjective+noun word combinations.

# 3. Data

The selection of an appropriate dictionary to extract the semantic information is our first task with potentially significant downstream impact. Dictionaries vary in construction methodology, definition-forming and presentation[7]. For the purpose of our proof-of-concept, we needed a resource that could be easily accessed and whose definitions can be easily parsed and mined for semantic primes.

Our solution uses Macmillan online dictionary as the main source of semantic information. The choice is informed, in part, by the dictionary’s construction as an aid for learners and thereby having definitions of a shorter length and comprising simpler words as well as useful usage tags ((e.g. offensive, impolite, British, American etc).

Further, Macmillan’s definitions are published online with embedded web links to definitions of selected component words. These helped to lighten the technical workload of mining the

[7] Čermák, F. (2011). Notes on Compiling a Corpus- Based Dictionary. Lexikos, 20(0). doi:10.5788/20-0-156 at https://www.ajol.info/index.php/lex/article/download/62737/50654

dictionary for the semantic components as the meaningful words appear to have been selected by the dictionary's publisher as significant towards clarifying the word's meaning. We found however, that these embedded links are only a subset of the component words that provide meaning, and the remaining component words were determined through POS tagging with UDPipe.

The disadvantages that have become evident during the process of semantic mining are linked to the quality of the definitions: some definitions explain the meaning through the words with a simpler semantic structure (**tell**: to give information to someone), while other describe a situation, usually through conditional statements: (**need**: if you need something, you must have it because it is necessary). The latter may overcomplicate the search for the related components.

To deliver a workable proof-of-concept within the short project timeframe,  we selected several words for mining and built a toy corpus stored in CSV format with semicolon as a separator. In our choice of words, we refer to Kruszewski and Baroni (2015)[8]. The corpus consist of 27 nouns and 29 adjectives. The nouns fall into the following classes: food, animals, people, abstract, places, things. The adjectives are selected to be compatible with single classes (e.g. *nutritious* is only compatible with *food*), or multiple classes (e.g. *clumsy* is compatible both with *people* and *animals*), some adjectives can be compatible with one word in the class but incompatible with another (e.g. *liquid milk* but not *liquid lemon*). The toy corpus was used in the development process.

65 semantic primes are stored as a dictionary as a part of a function and fed into the system when creating the word database which will be discussed in the next section.

# 4. Solution

The goal of our system is to generate word combinations and evaluate their correctness in terms of semantic compatibility. For this purpose, the distance between words should be measured. The distance between the words is obtained from the similarities and differences between semantic trees. The semantic trees are derived from the graphs representing the relationships between the semantic components forming the meaning of the word. The graph is constructed in the process of the iterative extraction of semantic components from the dictionary definitions. Before that, preliminary processing and dictionary scraping are applied.  The process is shown in Figure 1.

[8] Kruszewski, G. Baroni, M. So similar and yet incompatible: Towards automated identification of semantically compatible words. Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 964–969, Denver, Colorado, May 31 – June 5, 2015. 2015 Association for Computational Linguistics

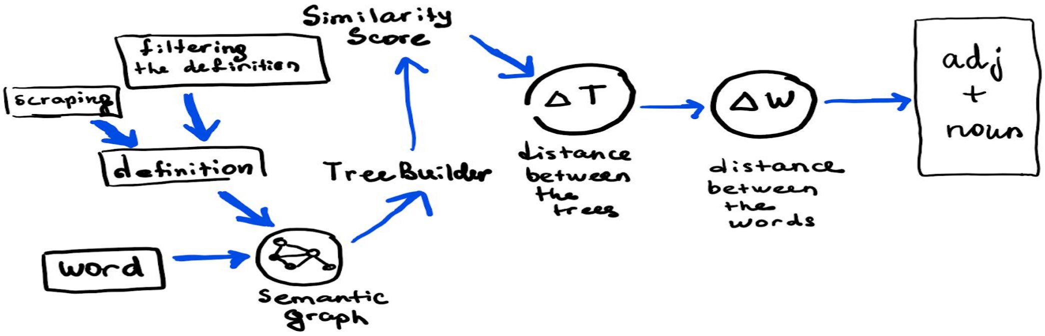


*Figure 1. The process of semantic mining*

**Step 1. Extracting the definitions from online dictionary. Dictionary scraping.**
Scraping refers to harvesting and selecting relevant information from the web. The definitions were first tokenized and lemmatized. In the dictionary definition, some words represent meaningful components - semes, while other, although frequent ones, store grammatical information or represent relationships (such as articles or prepositions). Thus, we used the POS (part of speech) criterion to extract the meaningful components. We used UDPipe for lemmatization and POS tagging, and selected only the words that received NOUN, ADJ, ADV, VERB, PART tags for further processing. Modifiers, for instance, negative particles, were considered important as well and have been taken into account. Lemmatization was needed in order to ensure that the different forms of the word are stored in the graph under the same lemma. We limited the definition parsing process only by the 1st meaning due to the time constraints, however, distinguishing homonyms and treating secondary and even figurative meanings would be important to fully represent the semantic structure with our approach.
Examples contained in the definitions and introduced by the corresponding phrases (for example, such as, e.g.) were eliminated, as well as usage indicators (impolite, offensive, etc).
The extracted components are passed to the semantic graph building function.

**Step 2. Building the oriented graph**
First, we build a primitive object which will be used for referring to semantic primitives in the process of building the graph. The nodes in the graph are word objects, both primitives and non-primitives, and the vertices are relationships between the words.
The algorithm uses breadth-first search to process the relevant words in the definition. Each word is added to the graph unless it already exists, in the latter case only the link is added. If the word appears in the definition of itself, it is ignored. There are three constraints to the search process: the depth of the tree, the appearance of the word the second time, and the word being a semantic prime.
The graph is stored in the TreeBuilder object.

**Step 3. Unrolling the graph into a tree. Pre-processing the tree for the distance computation.**
The search for the information necessary for determining the distance between the trees. The graph is explored with the breadth-first search. For each word, we count the number of occurrences and input it into step 4.

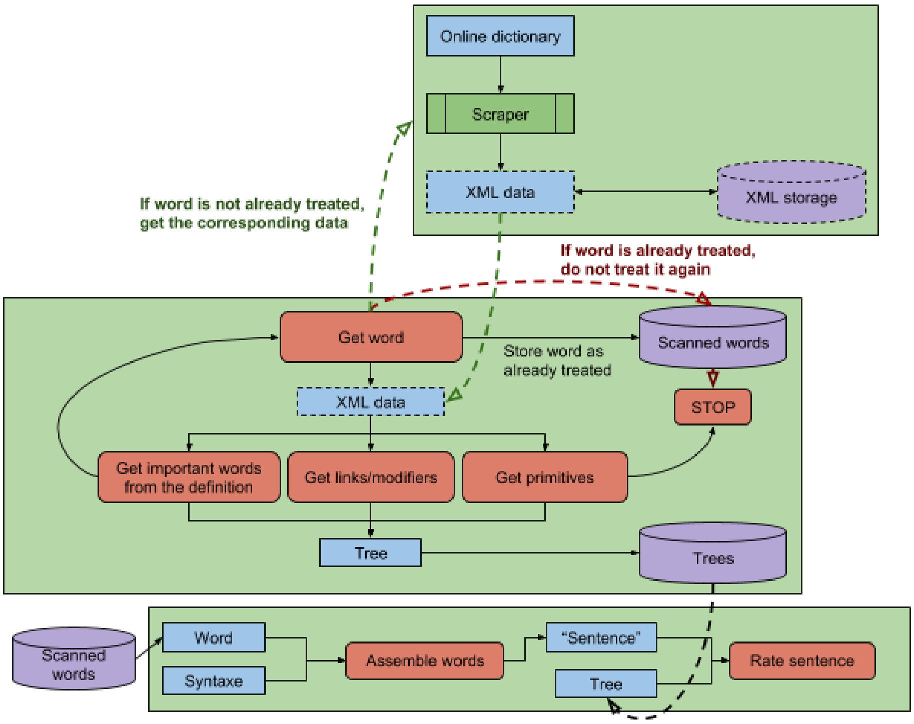
**Step 4. Distance between semantic trees**
The similarity score is determined according to the number of common occurrences of words: similarity = number of words shared by both trees / total number of words in the two trees.

The difference between the words is computed according to the following formula: difference = (number of words of the first tree not in the second one + number of words of the second tree not in the first one) / total number of words in the two trees.

### Step 5. Producing compatibility matrices

The compatibility matrices are difference and similarity matrices built for all the selected words. The higher the similarity score, the more compatible the words are, and vice versa for the difference score. There is no explicit word combination generation part, however, the compatibility score implicitly shows whether a certain word combination is possible.

Figure 2 summarizes the overall system architecture.



*Legend*

| | |
|---|---|
| Violet cylinders | Data storage |
| Green boxes | System modules |
| Red cells | Operations |
| Blue cells | Types of Data |
| Dashed arrows | Queries between parts of the system |
| Dashed cells | Planned features |

*Figure 2. Semantic mining system architecture*

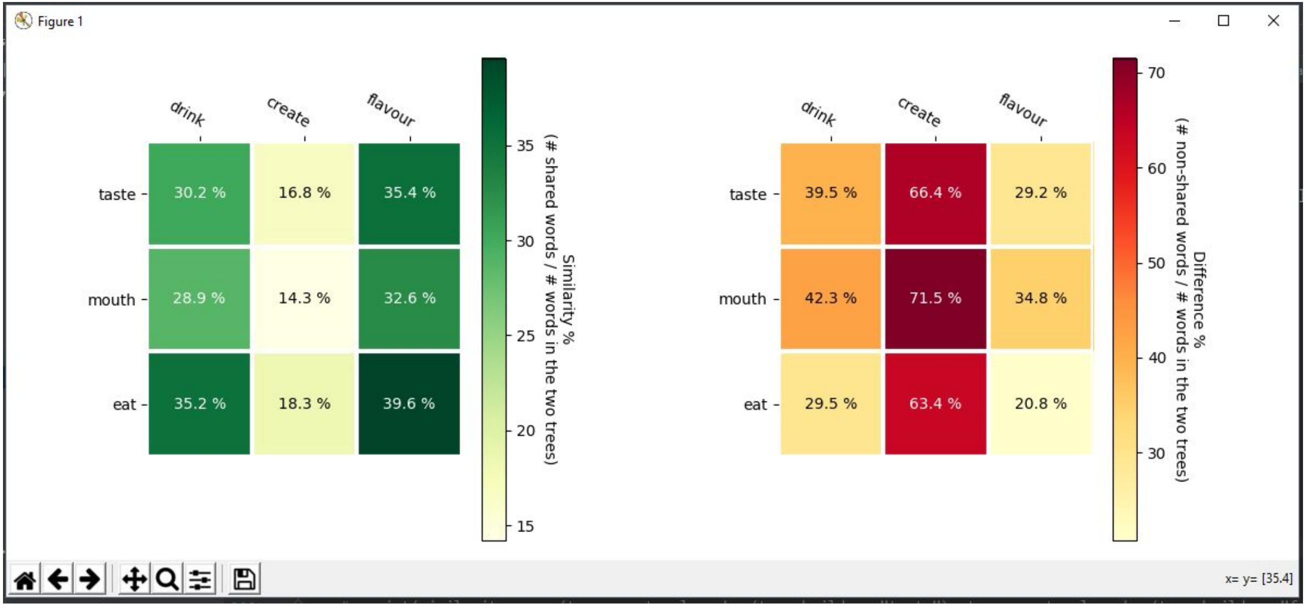
# 5. Usage

The step-by-step guide to the exploitation of our system is to be found as a supplement to this report and is available in the README section on git: https://github.com/EMarquer/semantic_trees/blob/master/README.md

## 6. Results

We obtained compatibility matrices for the nouns and adjectives stored in our toy corpus, which have proven our assumptions about compatibility issues. For instance, the similarity scores for obviously related words such as “lemon” and “sour” or “skilled” and “mechanic” were higher, while the same scores for unrelated words, e.g. “sweet” and “castle” or “happy” and “tree” are insignificant. The difference score shows the opposite trend.

The system was also run on randomly selected words. Figure 3 displays the corresponding compatibility matrices.



| | drink | create | flavour |
|---|---|---|---|
| taste | 30.2 % | 16.8 % | 35.4 % |
| mouth | 28.9 % | 14.3 % | 32.6 % |
| eat | 35.2 % | 18.3 % | 39.6 % |

| | drink | create | flavour |
|---|---|---|---|
| taste | 39.5 % | 66.4 % | 29.2 % |
| mouth | 42.3 % | 71.5 % | 34.8 % |
| eat | 29.5 % | 63.4 % | 20.8 % |

Figure 3. *Compatibility matrices: green – similarity score, red – difference score.*

All in all, the two scores – similarity and difference – seem equivalent and relevant: the words “eat” and “create” are completely unrelated and thus the similarity score is very low, while the difference score is the opposite, while for semantically closer words such as “eat” and “flavour” or “taste” and “drink” the scores are relatively higher. Nevertheless, the difference scores seem to be confusing as the rates for “mouth” and “drink” are still quite high, although we consider these words related. This exact interpretation of the difference in the scores, as well as generally the issue of compatibility according to the number of shared semantic components needs further investigation and exceeds the scope of our project.

Another observation that can be pointed out that the system is resilient to typos thanks to the embedded search engine in the dictionary

## 7. Conclusion

As a result of our project, a method to represent the semantic structure of a word, the system to obtain this structure from a dictionary, and the evaluation method have been developed.
The results show that our approach, besides having numerous limitations and requiring further development, demonstrates significant potential in handling the tasks of computational representation of the word meaning, estimating semantic differences and similarities between the words, and generating meaningful word combinations.

The revealed limitations fall into three categories: theory, data and algorithm related.

First, the theory of semantic primes need to be revised. The existing list of primes might realistically represent the cognitive aspect of meaning, which does not seem to be enough for computational purposes. We suggest expanding the list of primes and explaining their meaning by linking to world states, while other meanings can be represented through the composition of primes. A thorough investigation of compatibility rules is essential to improve the performance of the system and make it efficient for the tasks of disambiguation, translation, and sentence generation.

Second, the performance of our system largely depends on the choice of dictionary, however, none of the existing online dictionaries seems to be completely sufficient for the task. In many cases, the definitions are given through more and not less complex lexical entities and include too many detail, which make the iterative search of semantic components endless. Besides, in the selected dictionary often the cases of polysemy and homonymy are often indistinct, which can undermine the reliability of the semantic structure obtained.

Third, improvements in algorithm may include handling multiple meanings and increasing the processing speed.

Also, the limitations of our project are related to the time constraints. We were able to explore only the first meaning in the dictionary, however, secondary and figurative meanings would provide important results. In addition, the deep structure of the word meaning – semantic components mined from the context – could be investigated to obtain a fuller picture and represent the actual meaning of the word that eliminated the dictionary bias, which is an important task for lexicography and conceptual semantics.

# Bibliography

1. Čermák, F. (2011). Notes on Compiling a Corpus- Based Dictionary. Lexikos, 20(0). doi:10.5788/20-0-156 at https://www.ajol.info/index.php/lex/article/download/62737/50654
2. Firth, J.R. (1957). "A synopsis of linguistic theory 1930-1955". Studies in Linguistic Analysis: 1–32. Reprinted in F.R. Palmer, ed. (1968). Selected Papers of J.R. Firth 1952-1959. London: Longman.
3. Gaume, B. Hathout, N., Muller, P. 2004. Word Sense Disambiguation using a dictionary for sense similarity measure. DOI: 10.3115/1220355.1220528. Accessed at: https://www.researchgate.net/publication/228601536
4. Gaume, B. Venant, F., Victorri, B. Hierarchy in lexical organisation of natural languages. Denise Pumain. Hierarchy in Natural and social Sciences, 3, Springer, pp.121-142, 2006, Methodos Series.
5. Kruszewski, G. Baroni, M. So similar and yet incompatible: Towards automated identification of semantically compatible words. Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 964–969, Denver, Colorado, May 31 – June 5, 2015. 2015 Association for Computational Linguistics
6. Kuznetsova, E. Lexicologia russkogo yazyka [The lexicology of the Russian language], 1982.
7. Leaning meaning in natural language processing: Twitter megathread. https://medium.com/huggingface/learning-meaning-in-natural-language-processing-the-semantics-mega-thread-9c0332dfe28e?fbclid=IwAR29kCGUo08phytwf5OfTwLr5hXod1fi_gKThDzFRUYBV1lLt9h91alcTKQ