

Multilingual_WSD_Report

by Nikita Jain

FILE	MULTILINGUAL_WSD_REPORT.PDF (1.37M)		
TIME SUBMITTED	16-APR-2015 02:52PM	WORD COUNT	5544
SUBMISSION ID	529334219	CHARACTER COUNT	28835

ABSTRACT

Users spend a large amount of time in searching, translating, processing, analyzing the large amount of information available online, which is also growing at a very fast pace. This becomes very difficult for a human being to look for all the resources in a small period of time. So a user can leverage a computational machine to do these tasks in very less time. For this, machines need to have the understanding of natural or human language. So that mentions can be correctly mapped to the corresponding real world entities, to have more informative and intuitive web experience.

For making this human computer interaction possible, machines are made to learn how to recognize entities, how to disambiguate them, and link them to the mentions in the input document.

Chapter 1:

INTRODUCTION

User spend a large amount of time in searching, translating, processing, analyzing the large amount of information available online. Also, these information are growing at a very fast pace with constantly changing their format, even they are available in multiple languages, which is making difficult for a human being to look for all the resources in a small period of time.

With the increasing advances in machines' hardware and software technologies, we can leverage them to do these tasks in a small amount of time, so that a user can make better decisions rather than wasting time in doing operations over the information. So there's a need to make computational machine understand, disambiguate and relate these textual information present in natural or human language, just as a human mind does.

For making this human computer interaction possible, one of the major tasks is to identify these entities. An Entity can be anything which exists either virtually or physically. It can be a person, location, organization, date, anything.

So for a given input text some of the important tasks a computational machine need to learn, to extract and gain knowledge through the text are:

- Entity Recognition [1]
- Entity Disambiguation [4,7]
- Entity Linking [10]
- Entity related task extraction [12]

Entity recognition is the task of identifying the named entities present in the input text. For example, see the Figure 1.1 highlighted words and phrases.

राष्ट्रपिता **महात्मा गांधी** का पूरा नाम **मोहनदास करमचंद गांधी** था। हम उन्हें प्यार से बापु पुकारते हैं। इनका जन्म **2 अक्टूबर 1869** को **गुजरात** के **पोरबंदर** में हुआ। सभी स्कूलों और शासकीय संस्थानों में **2 अक्टूबर** को इनकी जयंती मनाई जाती है। उन्होंने प्रेरणा से हमारा देश **15 अगस्त 1947** को आजाद हुआ।

गांधीजी के पिता **करमचंद गांधी** **राजकोट** के दीवान थे। इनकी माता का नाम **पुतलीबाई** था। वह धार्मिक विचारों वाली थी।

उन्होंने हमेशा सत्य और अहिंसा के लिए आदोलन चलाए। **गांधीजी** वकालत की शिक्षा प्राप्त करने के लिए **इंग्लैंड** भी गए थे। वहाँ से लौटने के बाद उन्होंने **बंबई** में वकालत शुरू की। **महात्मा गांधी** सत्य और अहिंसा के पुजारी थे।

Figure 1.1: An example of Entity Recognition.

As shown in Figure 1.1, there are many mentions like ‘गांधी’ which refers to many persons like ‘सोनिया गांधी’, ‘राजीव गांधी’, ‘राहुल गांधी’ and many others. A human brain can easily disambiguate these kind of mentions, but a computational machine cannot. So there's a need to make a computational machine learn entity disambiguation, which deals with mapping mentions to their correct meaning.

There are millions of documents just for a single entity (E.g. Figure 1.2). So there's a need for linking these entities to have a good information source. So a machine needs to learn Entity Linking also.

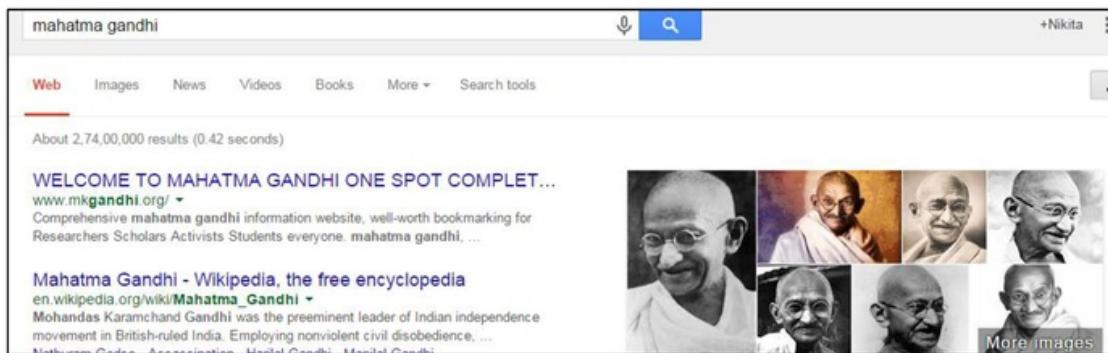


Figure 1.2: Search result for Mahatma Gandhi using Google Search Engine.

After having such informative resources, machine needs to be trained to extract knowledge using those resources. Some of the applications for entity oriented task extractions are query expansion, query prediction, trivia mining, building idiosyncratic knowledge bases and many others.

1.1 Outline of Report

Chapter 2 starts with an overview of named entity recognition and key term extraction. It continues to provide the challenges involved in NER, especially for idiosyncratic documents. It describes about the main features of NER, which can be used to mitigate the challenges in NER for idiosyncratic documents. As there can be multiple candidate entities for a particular mention, so there's a need to solve this ambiguity. Chapter 3 discusses about this problem and also explains about a unified approach to solve both EL and WSD. Chapter 4 focuses on the recognition and disambiguation task for verbs and verbal phrases using syntactic and semantic feature of input text. Chapter 5 covers the aspects of EL and an approach to handle one of the major challenges in EL, i.e. sparseness of knowledge base for unpopular entities. Given a brief introduction to NLP, IR and ML tasks, Chapter 6 focuses on the application part, where all the above techniques can be exploited to extract relevant tasks for a particular entity.

Chapter 2

NER FOR IDIOSYNCRATIC DOCUMENT

To reduce the burden of manual extraction of entities or concepts from a given document of specialized domain like, science, mathematics, there's a need to identify all the valid entities related to the given idiosyncratic document. This chapter will discuss about how to mitigate this problem.

Example:

Figure 2.1 shows the valid entities in the specialized domain PDF document, given as an input to the approach discussed in this Chapter.

arXiv:1405.7432 · Add annotation · ADS · INSPIRE

Thickness-shear Frequencies of an Infinite Quartz Plate with Material Property Variation Along the Thickness

Ji Wang, Wenliang Zhang, Dejin Huang, Tingfeng Ma, Jianke Du

Input

Materials Science

Properties of the quartz **crystal** blank of a resonator is assumed **homogeneous**, uniform, and perfect in design, manufacturing, and applications. As end products, quartz **crystal** resonators are frequently exposed to gases and **liquids** which can cause surface damage and internal degradation of blanks under increasingly hostile conditions. The combination of service conditions and manufacturing process including chemical etching and polishing can inevitably modify the surface of quartz **crystal** blanks with changes of material properties, raising the question of what will happen to vibrations of quartz **crystal** resonators of thickness-shear type if such modifications to blanks are to be evaluated for sensitive applications. Such questions have been encountered in other materials and structures with property variations either on purpose or as the effect of environmental or natural processes commonly referred to as functionally graded materials, or FGMs. Analyses have been done in applications as part of studies on FGMs in structural as well as in acoustic wave device applications. A procedure based on series solutions has been developed in the evaluation of frequency changes and features in an infinite quartz **crystal** plate of AT-cut with the symmetric material variation pattern given in a cosine function with the findings that the vibration modes are now closely coupled. These results can be used in the evaluation of surface damage and corrosion of quartz **crystal** blanks of resonators in sensor applications or development of new structures of resonators.

Output

Vibration, Acoustic wave, Homogenization, Liquids, **Crystal**, Resonance, Frequency, Materials

Figure 2.1: Materials Science text as input and its mentioned entities as output.

Key Terms:

Valid Entity: It is an N-gram string representing a relevant concept with respect to the specialized domain of the given document, not just any real world object.

E.g. Division in which the fractional part (remainder) is discarded is called **Integer** division.

[**Integer**: In a pure mathematical sense, it's an entity, whereas in other domains it is just a linguistic construction]

Named Entity Recognition: It is the task of recognizing all the entities (such as name, location, organization, thing) mentioned in the given input text.

E.g. **Newton** built the first practical reflecting telescope.

[**Newton**: English physicist and mathematician name]

Key Term Extraction: It is the task of automatic extraction and ranking of relevant terms present in the given input text.

E.g. History of IIT Roorkee is taken as input text and passed through an online key term extraction tool, '**maui-indexer**', which gives the frequent key terms an output, as shown in Figure 2.2.

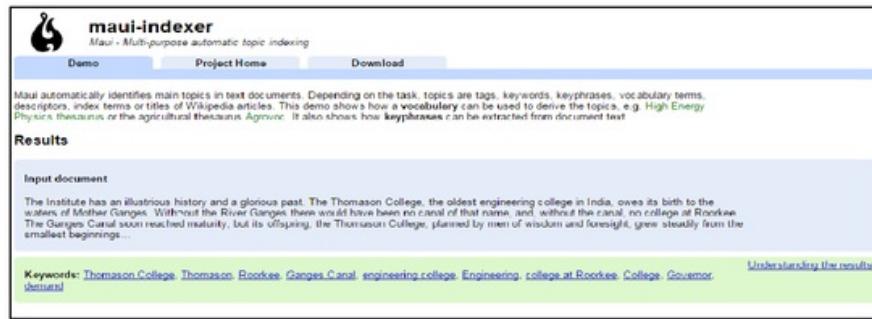


Figure 2.2: Extracting key terms (key words) from the input document.

Challenges:

Identifying entities or concepts in idiosyncratic collections are difficult because of the

- Nature of content (like Novelty, Specificity).
 - Scarcity of structured or semi-structured information for specialized domains.

Approach:

The approach being discussed below lies in between key term extraction and named entity recognition problem, as it gives only those entities or concepts as an output which are relevant to the given idiosyncratic document.

The strategy followed is to extract all the possible N-grams from the input document, and then inspect the document on the basis of semantic and syntactic features like Part of Speech, N-gram co-location statistics, general and specialized knowledge base.

Procedure in broader terms:

1. Extraction of text from the input document.
 2. Data preprocessing (Lemmatization).
 3. Multi-step N-gram candidate generation.
 4. Select N-grams from step 3, based on the predefined syntactic and semantic features.
 5. Using supervised classifier, decision tree to generate the final ranked list of N-grams, which reflects the valid entities or concepts with respect to the given document.

The steps are explained with the help of examples shown below:

Input: PDF document shown in Figure 2.3



Figure 2.3: Scientific document in PDF format.

Given: DBpedia, DBLP, ScienceWISE

Step 1: Text extraction

On the receiving PDF document, convert it into raw text using an open source library (Apache Tika)

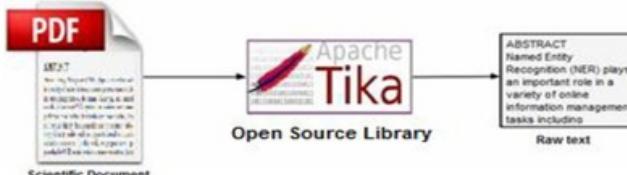


Figure 2.4: Content (.PDF) converted into Content (.CSV) using Apache Tika.

Step 2: Data Preprocessing

Following preprocessing operations are to be performed over the raw text

- i. Lowercase all the words (except acronyms) in the raw text.
- ii. Maintain two copies of the above text, one is passed to the POS tagger and another is passed through lemmatizer.
- iii. Build an N-gram index list of the lemmatized copy of the text.

Step 3: Multi step N-gram candidate generation

This step is based on word co-locations.

- i. Extract all the bi-grams from the N-gram index list having a frequency greater than a predefined threshold.
- ii. Now, the above bi-grams are merged together into a valid tri-gram having a frequency greater than a predefined threshold.
- iii. Repeat this process for tri-grams, up to N.
- iv. Apply frequency re-weighing process (as shown in Figure 2.5) on N-grams, which appears as a part of other N-grams in the indexed list after Step 3.iii.

N-gram	Frequency		N-gram	Re-weighted Frequency
natural language	63		natural language	0
language processing	63		language processing	0
natural language processing	63		natural language processing	63

Figure 2.5: Showing frequency re-weighing process.

Frequency re-weighing process is applied as some N-grams occurs as a subpart of other N-grams (see Figure 2.5), which can't occur as a separate entity, but only as a bigger entity. So the subpart N-gram frequency is changed to 0.

- v. Eliminate all N-grams having re-weighted frequency equal to zero.
So from Figure 2.5, 'natural language' and 'language processing' N-grams are removed.

Step 4: Select N-grams obtained from Step 3, based on the predefined syntactic and semantic feature space. It consists of following features to detect named entities in scientific documents:

i. **Part of Speech Tags**

Several POS tags patterns are grouped into related patterns to reduce the feature space, where each POS tag pattern will represent a separate binary feature.

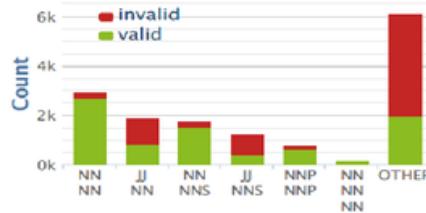


Figure 2.6: Top six most frequent POS tag patterns of the SIGIR collection, where (JJ: Adjectives, NN: Singular Noun, NNS: Plural noun, and NNP: Proper Noun) (courtesy: [1]).

Using results like in Figure 2.6 [1], based on the specialized domain document (Information Retrieval in Figure 2.6), several features are designed which may perform better than predefined and common for all POS tag patterns.

ii. **Near N-Gram Punctuation**

- The presence (+punctuation) of punctuation mark before/after an N-gram has been almost twice as often for the valid N-gram to invalid N-gram.
- And the absence (-punctuation) of punctuation marks before/after an N-gram is less frequent for the valid N-gram than invalid ones.

iii. **Domain Specific knowledge base**

As knowledge base stores high quality information (keywords) which are used by authors in their publications, hence they can be used as a highly discriminating feature.

Knowledge Base for Computer Science Domain:

DBLP

Knowledge Base for Physics Domain:

ScienceWISE

iv. **General Purpose Knowledge base**

Wikipedia/ DBpedia (machine readable Wikipedia) features help in identifying valid entities or concepts as:

- Most of the pages in Wikipedia contain valid entities.
- Interlink of relevant concepts through links in the page body and their categories.
- Alternate spelling pages are also connected to each other through ‘redirect’ property.

Using following NER features to exploit the relation graph build from the Wikipedia data:

- Check for the match between N-gram candidate and Wikipedia page title
- Check for the match between N-gram candidate and the alternative spelling of a Wikipedia page,
- Give weights according to the size of the connected N-gram components (with and without the redirect property),
- Give binary feature weight as 1 when an N-gram component is connected with at least one DBLP keyword and 0 otherwise;
- Give weights according to the number of outgoing links from the Wikipedia page to other Wikipedia pages.

v. Syntactic Feature

Some of the syntactic features which can be used to identify the entities are

- Length of N-gram component,
- Casing (Upper is good) of the N-gram component,
- Count of the superset of N-grams (present in the input document) containing the given N-gram as their subpart.

Step 5: Using the above feature space and few manually extracted entities from scientific documents are used to train the classifier (decision tree) to effectively identify all the valid entities or concepts from the given document.

Facts or Assumptions:

1. In this approach more focus is on the identification of N-grams entities with $N > 1$. As most of the uni-gram entities are very generic in nature, and hence they can be easily recognized with the help of a dictionary.
2. General purpose knowledge base used in this approach is DBPedia (contains all the entities of Wikipedia) and domain specific knowledge base are DBLP (for Computer Science concepts) and ScienceWISE ontology (for Physics concepts).
3. DBLP (Digital Bibliography and Library Project) provides high quality metadata (keywords that authors assign to their publication) and links to the research articles and publications.
4. ScienceWISE Ontology contains concept discussed in research papers in physics, available at ArXiv.org (the standard all physics preprint server), representing valid named entities.

Chapter 3 Named Entity and Word Sense Disambiguation

To reduce and combine the large amount of efforts being put up in the research to disambiguate the context and entities, a unified approach is being discussed in this chapter. The goal is to disambiguate the mentions or fragments (entities or concepts) present within a text and to link them to the most suitable entry in a reference inventory.

Example:

Figure 3.1 shows an example of entity linking (Obama and States) and word sense disambiguation (Rock Concert) using Babelfy, a multilingual tool for word sense disambiguation and entity linking.



Figure 3.1: Given an input text and its disambiguated fragments.

Key Terms:

Entity Linking: It is the textual mention which can be linked to a named entity and it may contain the partial mention. Entities are linked with the most suitable entry using reference knowledge base.

Eg. **Modi** was there to receive **Obama** at **IGI Airport**, Delhi.

[Modi: *Narendra Modi*; Obama: *Barack Obama*; IGI Airport: *Indra Gandhi International Airport*]

Word Sense Disambiguation: It is the task of choosing the right sense for a word within a given context. The mentions are complete here and they are disambiguated using Dictionary or Lexicographic knowledge.

Eg. **Bank** cuts fixed deposit rates by up to 0.25 per cent

[Bank: *Financial institution*]

Dense Heuristic Graph Algorithm: It solves a problem of finding dense subgraph of maximum density by applying some heuristics to speed up the process of finding a satisfactory result, as dense graph finding algorithm is NP-Hard in nature.

Approach:

This approach gives a unified knowledge approach to solve both problems, Word Sense Disambiguation and Entity Linking.

The strategy followed is to automatically create edges between the other possible senses for the mention by assigning weights to the existing semantic directed network and then performing Random Walk with Restart on this network. This way the semantic signature for all the mentions in the input text is created.

The steps are explained with the help of examples shown below:

Input Text: Obama went to Rock Concert in States

Given: BabelNet (Lexicalized and encyclopedic multilingual semantic network)

Step 1: Mentions identification is done as

Input Text → POS Tagger → Mentions/Fragments

Mentions/Fragments:

- Sequences of words of maximum length 5
- Contains at least one noun
- Substring of lexicalizations in BabelNet

POS Tagger: Here Stanford Part of Speech (POS) Tagger is used, any other POS tagger can also be used

Here our mentions are {‘Obama’, ‘Rock’, ‘Concert’, ‘Rock Concert’, ‘States’}. This step is performed just once and it is independent of the relation between the input text’s mentions.

Step 2: Using BabelNet we find the candidate meaning for all the mentions found in Step 1, as shown in Figure 3.2.

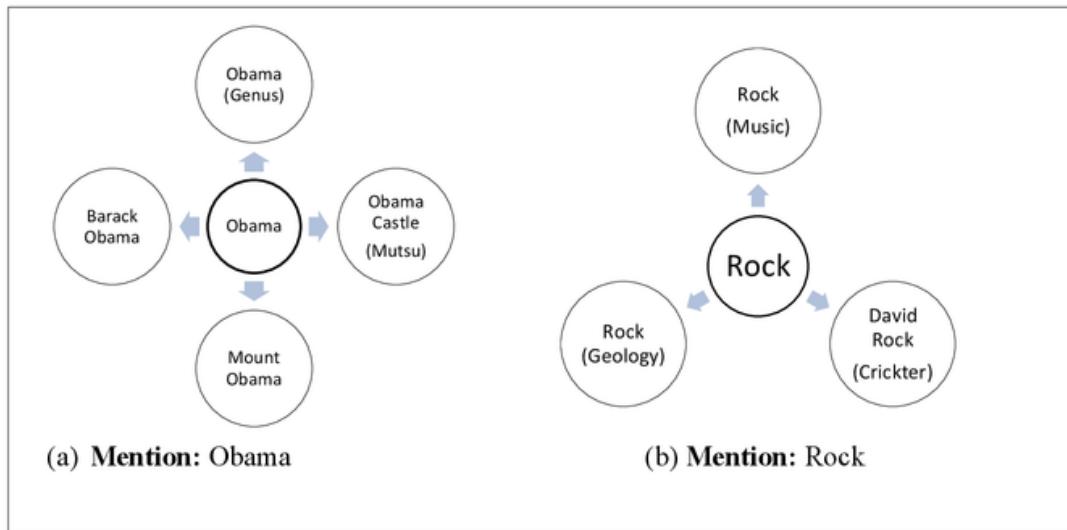


Figure 3.2: Mentions and their possible candidates for semantic signature (continued)

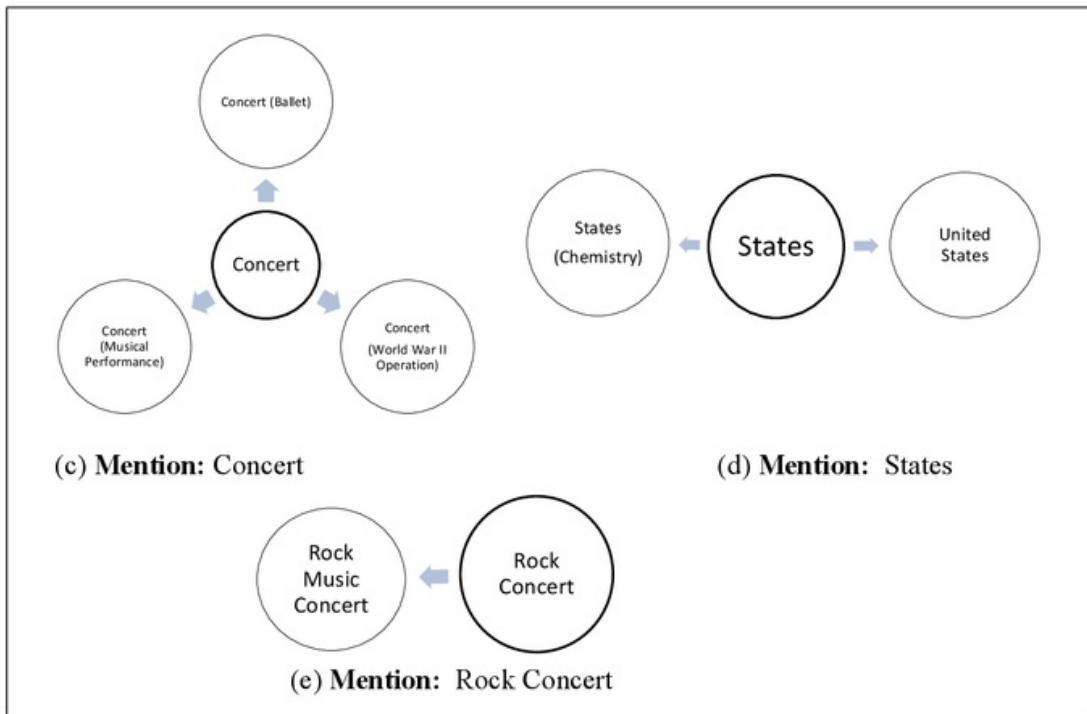


Figure 3.2: Mentions and their possible candidates for semantic signature.

Step 3: Create a directed graph $G(V, E)$ (illustrated in Figure 3.3), such that

V : Set of all the mentions along with their possible meanings.

E : Set of all the edges connecting the related meaning vertices such that

- Two same mentions should not be joined together.
- Destination vertex's mention should be candidate signature of the Source vertex's mention.

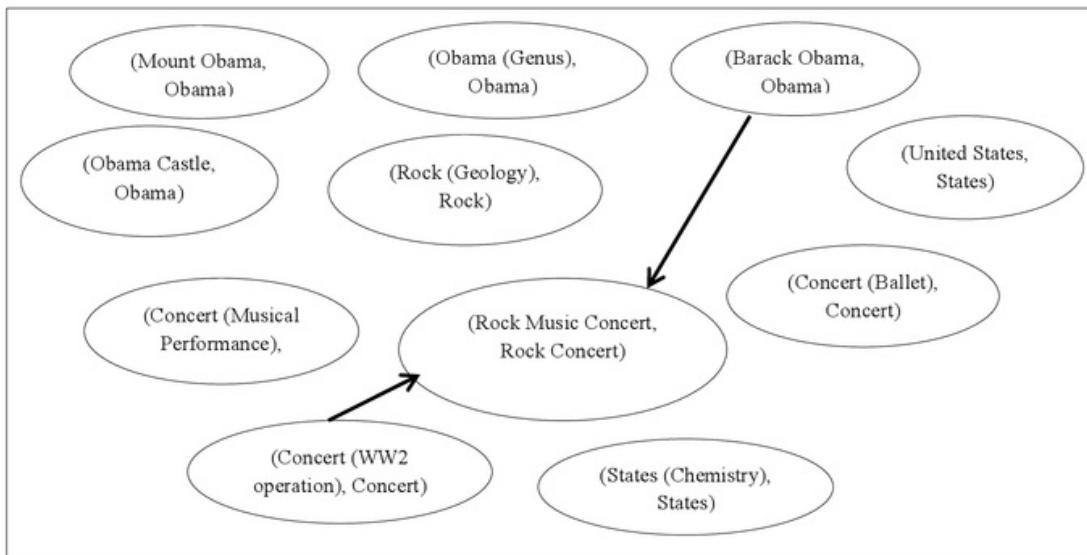


Figure 3.3: Intermediate semantic interpretation of the input text.

Step 4: Using the graph G (V, E) generated in the step 3, reduce its degree of ambiguity using densest subgraph heuristic. Hence, only the most suitable meaning of each mention will belong to its densest area of the graph.

Step 5: Using the graph G (V, E) generated in the step 4, select the most suitable meaning for each mention using predefined threshold. For the given input text its one to one mapping is shown in the Figure 3.4.

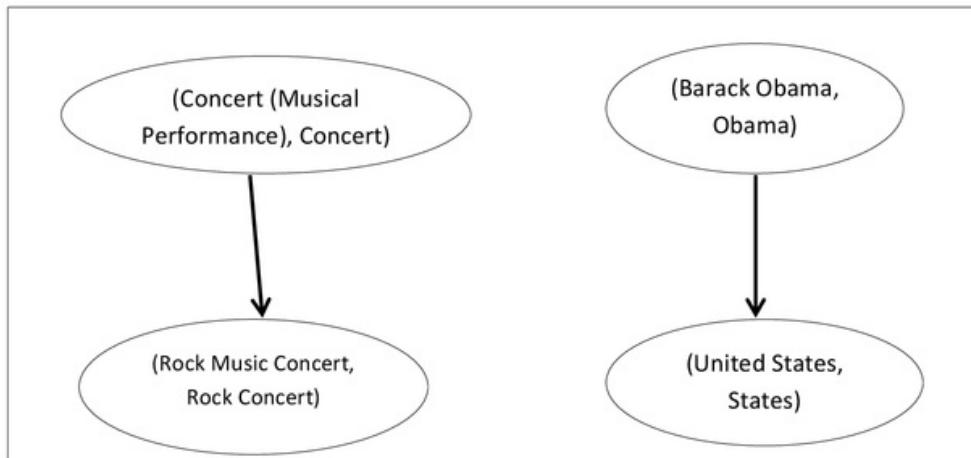


Figure 3.4: Final semantic interpretation of the input text.

Some Multilingual Examples:

(A) Input Language: Hindi



Figure 3.5: Text [‘The posters of missing Advani were posted in Gandhinagar, Gujarat’] and its disambiguated mentions.

(B) Input Language: French



Figure 3.6: Text [‘Sophie Marceau likes to wear gold jewellery’] and its disambiguated mentions.

Facts or Assumptions:

1. This approach can be used for multilingual settings.
2. BabelNet is a large multilingual knowledge graph built up from various encyclopedias and lexicographic dictionaries. Other knowledge graph with such structural and lexical capabilities can also be used.
3. It is assumed that knowledge used in word sense disambiguation is also useful for entity linking task and vice versa.
4. The only information used from the knowledge graph is the vertices, while neglecting the relation type connecting the vertices.

Limitations:

Since this approach is based on the accuracy of BabelNet and Standford POS Tagger, which at times generate wrong results, leadig to wrong or incomplete mention meaning selection.

Proposed Solution for this problem:

Use multiple knowledge graphs and POS tagger tools, and all of their results to decide the next step to take up.

Chapter 4

Verbs and Verbal Phrases: Recognition and Disambiguation

To completely disambiguate the sense of natural language text, there's a need to disambiguate the verb and verbal phrases along with noun phrases in the input text and to increase the recall and precision of existing WSD methods.

Example:

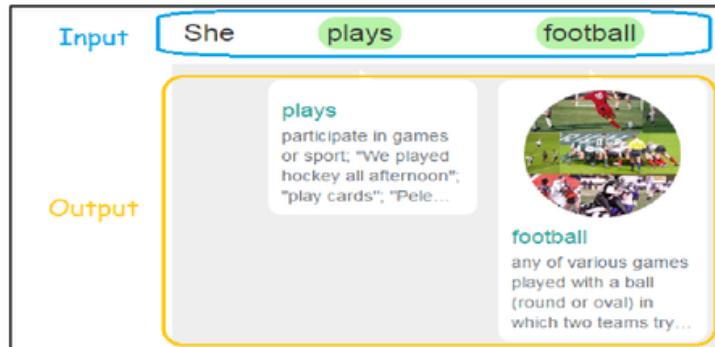


Figure 4.1: Given an input text and its disambiguated fragments.

Key Terms:

4

Verbal Phrase: It is a syntactic unit composed of at least one verb and its dependent (objects, complements and other modifiers), but not always including the subject.

E.g. She **was walking** quickly to the mall.
[Walking: *The act of travelling by foot*]

Dependency Parsing: Syntactic structure consists of lexical items, linked by binary asymmetric relations called dependencies. This type of parsing which looks for the grammatical structure of sentences to find the connection between the words is called dependency parsing. An example is shown below in Figure 4.2, depicting the dependency structure of an input text using Stanford NLP Parser.

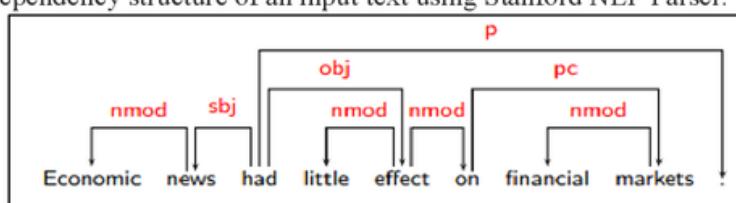


Figure 4.2: An example of dependency parsing.

Approach:

This approach has discussed about a framework for word recognition and disambiguation (Werdy) which deals with:

- Verbs and Verbal Phrases
- Multiword expressions (E.g. Take a breath)

The strategy followed is to prune the verb candidate sets syntactically and semantically. So the non-matching verb senses gets pruned before passing it to state-of-the-art general WSD method to select the most suitable sense. Hence accuracy and efficiency of WSD task are significantly improved.

Procedure in broader terms:

1. Recognise all entries present in the input text.
2. Do syntactic pruning.
3. Do semantic pruning.
4. Pass it through Word Sense Disambiguation method.

The steps are explained with the help of examples shown below:

Input Text: Albert Einstein remained in Princeton.

Given: WordNet, VO Sense Repository

Step 1: Entry Recognition

Find the set of candidate senses of all the entries (single/multi words) found in the knowledge base (WordNet) along with their part-of-speech-tags. This is done by generating the dependency parse tree for every entry (lemmatized), and then matches it against the knowledge base.

Here are the entries {'Albert Einstein', 'Remain' and 'Princeton'}

Here our focus is on the verb sense disambiguation, so only verb senses and their disambiguation related tasks are performed.

In the given input text the only verb present is 'Remain' and Figure 4.3 is showing the all the possible candidate sense for it. Our goal is to select one of the candidate senses.

WordNet Search - 3.1
- [WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for:

Display Options:

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (gloss)

Verb

- S: (v) [stay](#), [remain](#), [rest](#) (stay the same; remain in a certain state)
- S: (v) [stay](#), [stay on](#), [continue](#), [remain](#) (continue in a place, position, or situation)
- S: (v) [remain](#) (be left; of persons, questions, problems, results, evidence, etc.)
- S: (v) [persist](#), [remain](#), [stay](#) ([stay behind](#))

Figure 4.3: WordNet glosses for verb (Remain) enclosed inside blue box.

Step 2: Syntactic Pruning

As there can be multiple verb senses for a single verb (see Figure 4.3), so there's a need to prune some of them. One of the ways to prune is according to the syntax of the input text.

Using WordNet frame⁶ (like ‘somebody verb something’) convert verb senses into seven different categories of clauses: SV, SVA, SVC, SVO, SVOO, SVOA, SVOC

Where S: Subject, V: Verb, A: Adverb, C: Complement, O: Object

For each candidate's sense of the mention verb, look for its clause type nature and match it with the mentioned verb in the input text. All non-matching candidate senses are pruned.

- 1 • S: (v) stay, remain, rest (stay the same; remain in a certain state) SVC
- 2 • S: (v) stay, stay on, continue, remain (continue in a place, position, or situation) SVOA
- 3 • S: (v) remain (be left; of persons, questions, problems, results, evidence, etc.) SV
- 4 • S: (v) persist, remain, stay (stay behind) SV

Figure 4.4: Verb (Remain) candidate sense along with their clause category.

Here in our Input text: “Albert Einstein remained in Princeton”, remained (verb) clause type is SVA. So we prune the candidate sense (1) because its sense is for clause category: SVC (shown in Figure 4.4)

Step 3: Semantic Pruning

After the syntactic pruning, we apply semantic pruning on the remaining candidate sense by focusing more on the subject of the verb, whether it is people or location or thing.

Using VOS Repository we remove the verb candidate senses whose semantic argument is not matching with the sentence semantic nature.

Eg. The man plays football

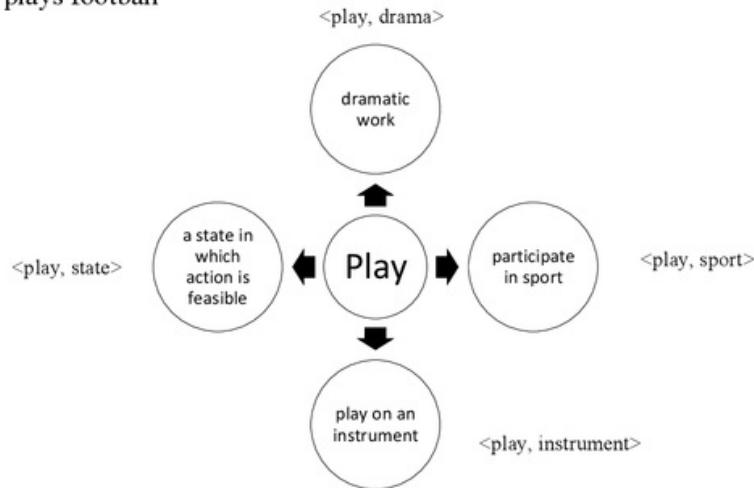


Figure 4.5: Semantic interpretation of verb candidate sense.

We can see in Figure 4.5 that <play, football> matches to only <play, sport>, so other verb candidate senses {<play, drama>, <play, state>, <play, instrument>} are pruned.

Step 4: Using all the remained candidate senses for all the mentions in the input text, we disambiguate their meaning using some standard word sense disambiguation algorithm.

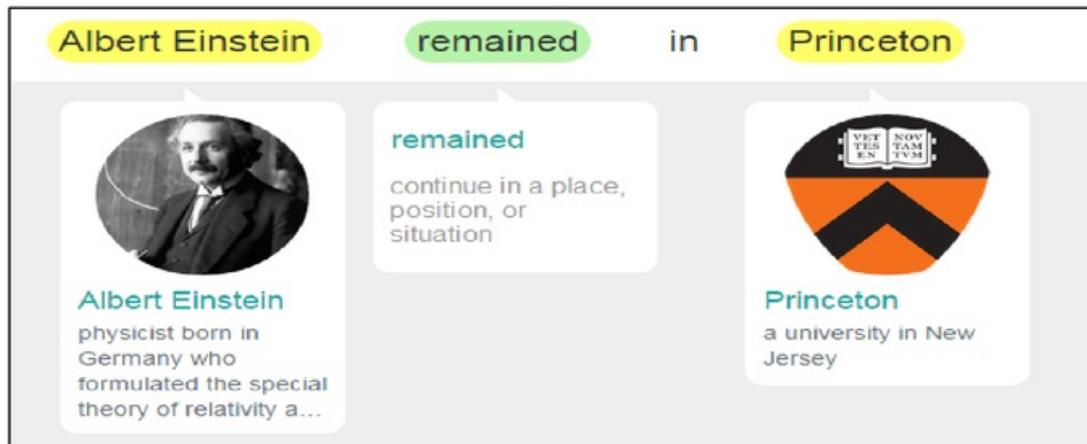


Figure 4.6: Final disambiguated version of the input text.

The output of this approach will be like as shown in the above Figure 4.6, that is mention (it can be noun or verb) and it's associated disambiguated meaning.

Facts or Assumptions:

1. This approach can be used for multi word entries.
2. WordNet is a large lexical database of monolingual language, built up from various thesaurus and lexicographic dictionary.
3. VOS Repository is a semantic Repository. Constructed from WordNet gloss-tags corpus, SemCor dataset, and manually created VO sense pairs.
4. To handle the incompleteness to some extent, hyponyms of the object-argument senses in VOS Repository is considered.

Chapter 5

Entity Linking: Using Random Walk with Restart

The main focus of this Chapter is to overcome the sparsity issue in entity linking for unpopular entities, as lexical and statistical features are very less for them. Also to correctly link the mentions, which are dissimilar to match (semantically) to their respective entities.

Objective:

The aim is to link all the mentions of entities in the given text document to their respective disambiguated entities using semantic similarity measure.

Example:

Figure 5.1 shows the disambiguated entities in the given input text, in English language

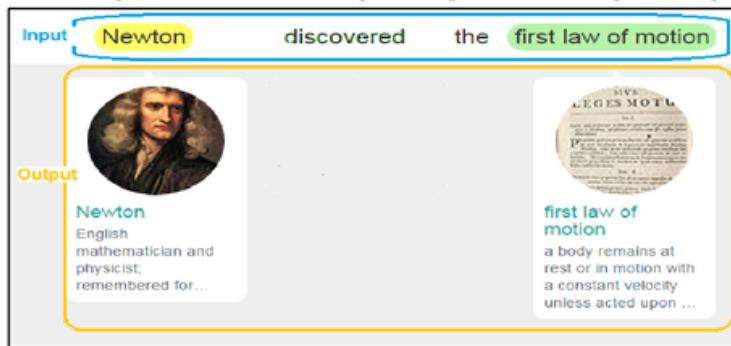


Figure 5.1: Given an input text (English) and its disambiguated fragments.

Key Terms:

Semantic Similarity: It is the measure defined over the set of documents or text to check how similar one concept is to another.

E.g. Jaguars are the largest of South America's big cats. [1]
[Jaguar: Cat is a hypernym of Jaguar]

Semantic Signature: It is a vector of mention containing all semantically related entities, which have the probability to occur with the mention.

E.g. Semantic Signature ('Gandhi') = {'Rajiv Gandhi', 'Sonia Gandhi', 'Mahatma Gandhi', 'Rahul Gandhi'}

Random Walk with Restart: It is the iterative process of finding the relatedness or proximity among two nodes by leveraging the global structure of the network. In this graph traversal the walker can jump back to the starting node with some predefined probability. It is like a finite automata having probabilistic action.

Approach:

The strategy followed to disambiguate the mentions, is by using the semantic signature of both entities and document. The semantic signature of the document is computed every time a new mention is disambiguated, so that the search is always made in the context.

This approach is greedy in nature. It is performed in two steps:

i. Local Method

Here disambiguation is done by using lexical features of the co-location words or mentions in the given document which are similar to it. And then rank the candidate entities in the set for all the mentions using some heuristics like most popular entity or any other to link that mention with the top most entity.

ii. Global Method

Local method is not enough to disambiguate the mentions because of the feature sparsity problem of the mentions. Also, there are some (Mention - Entity) pair which are not semantically related, so they will be missed by the local method.

E.g. ('Sachin Tendulkar' – 'God of Cricket')

So there's a need for global method which will take care of the unpopular mention and entity linkage. Also, it takes care of the fact that disambiguation of one mention will help in the disambiguation of other mentions present in the document.

Procedure in broader terms:

1. Create a semantic signature for all the mentions in the document using some dictionary (E.g. Wikipedia pages).
2. Using semantic signatures created in Step 1 to disambiguate all the mentions.

The steps are explained with the help of examples shown below:

Input: A document with all the mentions being marked, shown in Figure 5.2.

The [IIT] have a common admission process for undergraduate admissions. It was called IIT-JEE, which was replaced by Joint Entrance Examination Advanced in 2013. The post-graduate level program that awards M.Tech degree in engineering is administered by the older IITs (Kharagpur, [Bombay], Madras, Kanpur, Delhi, Roorkee, Varanasi, Guwahati). M.Tech admission is done on the basis of Graduate Aptitude Test in Engineering (GATE). In addition to B.Tech and M.Tech programs, [IIT] also award other graduate degrees such as M.Sc in Engineering, Maths, Physics and Chemistry, MBA, PhD etc. Admission to these programs of [IIT] is done through Common Admission Test(CAT), Joint Admission Test for Masters (JAM) and Common Entrance Examination for Design (CEED).

Potential tags:
LOCATION
ORGANIZATION
DATE
MONEY
PERSON
PERCENT
TIME

Figure 5.2: Wikipedia Page on IIT passed through Stanford NER tool.

Given: A Knowledge Base (in graphical format), Dictionary

Step 1: Create an entity graph using the candidate entity set (find from a knowledge base) for all the mentions present in the document and interlink all the related entities, as shown in Figure 5.3.

* For some of the mentions present in the document entity graph is shown in Figure 5.3

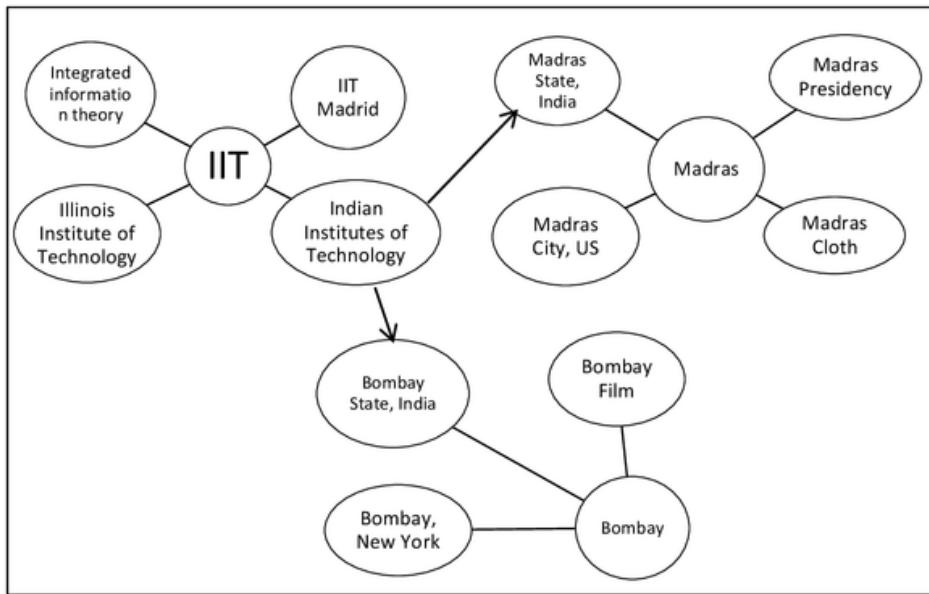


Figure 5.3: Entity graph for the input document (using only three mentions).

Each node (candidate entities) is assigned a stationary probability using random walk with restart, which indicates the relatedness of the node in the graph.

Step 2: Find semantic signatures for all the candidate entities, of all the mentions using random walk with restart over their graphs, created in Step 1.

Step 3: Find the semantic signature for the document using random walk with restart over the graph having only disambiguated mentions as its node. This Step is performed every time a new mention is being disambiguated.

Step 4: This step is performed for all the mentions present in the document using,

- Semantic signature of all the candidate entities,
- And semantic signature of the document found in Step2 and Step 3

They are used to calculate the similarity score for all the candidate entities.

Step 5: This Step is also performed for all the mentions present in the documents

- Using similarity scores generated for each candidate entity rank them,
- Then select the top most candidate entity having score more than predefined threshold for the mention it belongs to.
- If there is no such candidate entity having a similarity score above the threshold, then the mention is linked to NIL, indication no good entity for it.

Step 6: Recompute the Step 3 if and only if, new mention are being disambiguated in Step 5. This approach will terminate when all the mentions are being linked to one of its candidate entity.

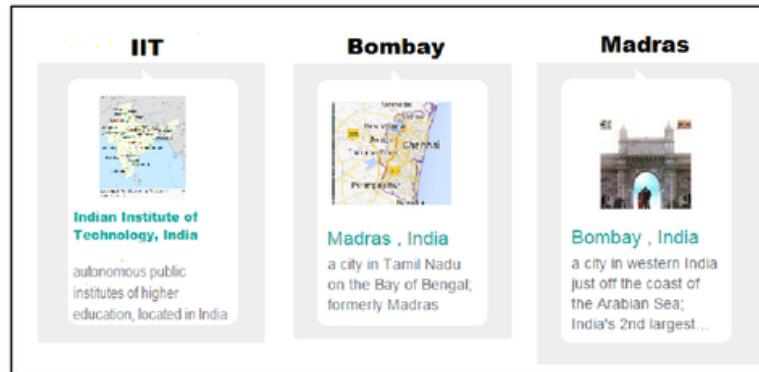


Figure 5.4: Disambiguated mentions of the input document.

Facts or Assumptions:

- i. Zero Kullback Leibler Divergence is used for measuring the semantic relatedness.
- ii. The dictionary is created from the Wikipedia titles, hyperlinks in a page body, redirection pages, and disambiguated pages.
- iii. The probabilities being assigned to the nodes, indicating their relatedness to the graph in which they are present.

To understand the user's goals, basically the tasks they look up to, we need to identify the tasks associated with the user/entity. The approach being discussed in this Chapter uses search query logs as a source document. This kind of entity oriented task extraction will help in improving the query prediction and query expansion.

Example:

Figure 6.1 shows an application where entity oriented task extraction is helpful.

(a) Screenshot of the Google search bar showing the query "where is taj mahal".

(b) Screenshot of the search results page showing related non-entities like Agra Fort, The Oberoi Amarvilas, Agra Hotel, ITC Mughal, and Hotel Panna Paradise.

(c) Screenshot of the search results page showing related entities like Taj Mahal, Agra Fort, The Oberoi Amarvilas, Agra Hotel, ITC Mughal, and Hotel Panna Paradise.

Figure 6.1: (a) Query made by the user; (b) Related non-entities with the entity 'Taj Mahal'; (c) Related entities with the entity 'Taj Mahal'.

Key Terms:

Query Term Prediction: Prediction made before the query begins, so that the user can decide on whether or not to run the query. It enables them to make better decisions and reduce the server load. As we can see on the bottom page of the google search result, a snapshot is shown in Figure 6.2 for the query 'Taj Mahal'.

Searches related to taj mahal

taj mahal history	taj mahal the true story
taj mahal wallpaper	taj mahal facts
taj mahal story	taj mahal timings
taj mahal information	taj mahal songs

Goooooooooooooogle >
1 2 3 4 5 6 7 8 9 10 Next

Figure 6.2: An example of query prediction on Google search engine.

3

Query Expansion: Query Expansion is the term given when a search engine adds search terms to a user's weighted search. The goal is to improve precision and/or recall. See Figure 6.3 for an example.

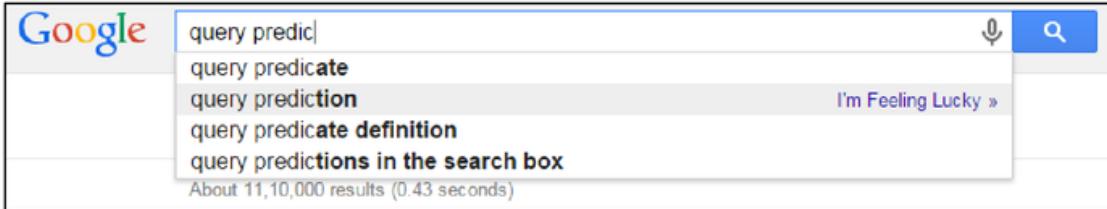


Figure 6.3: An example of query expansion on Google search engine.

Approach:

2

This approach discusses about entity based task extraction method, to create a comprehensive list of the tasks associated with an entity.

The strategy followed is to rank/score all the non-entity terms attached to the category. And then choosing some predefined number of the top results to further predict or expand the search query.

Procedure in broader terms:

1. Group all the related entities and their respective nonentities terms under one umbrella, i.e. category
2. Using the above step result construct task dictionary.

The steps are explained in details with the help of examples shown below:

Input Text: Flights between Paris and London.

Given: Dexter (Open Source framework for Entity Linking)

Step 1: Using Dexter find out the entities and their categories in the search query made

Entity	Category
Paris	City
London	City

Step 2: Aggregate all the related entities under one umbrella, i.e. Category.

Entity	Category

2

{Paris, London}	City
-----------------	------

Step 3: For each entity in the query, associate the non-entity terms with its category.

Entity	Non-Entity	Category
{Paris, London}	{Flight}	City

Step 4: Filter and rank the list of each category containing several terms (obtained in the above step).

The ranking of these terms is done using a tf - Idf method, such that

5

$$\text{tf-Idf}(t,c) = \text{tf}(t, c) * \log(N_c / N_t)$$

Where,

$\text{tf}(t, c)$: Frequency of term t in category c

N_c : Total number of categories

2

N_t : Number of categories that contain the term t

Step 5: Prune all those words in the category lists that are below a certain threshold.

See an example (shown in Figure 6.4) for the entities ‘wall panel’, ‘wood furniture’ and ‘carpentry’.

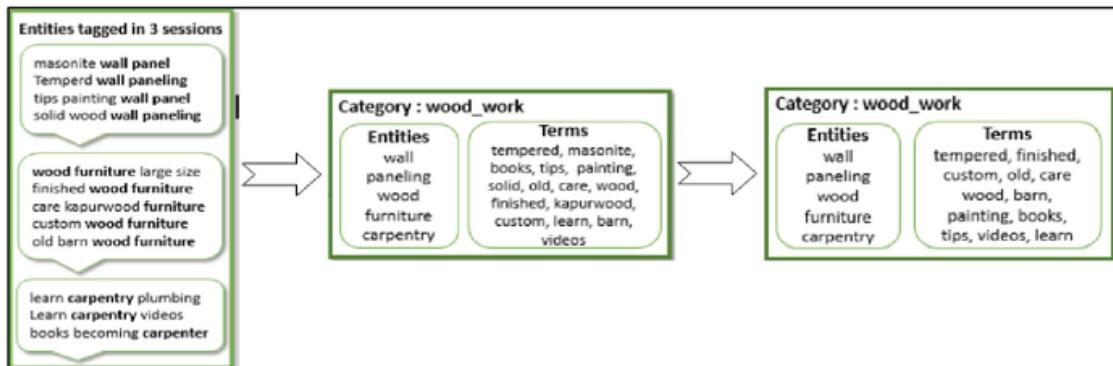


Figure 6.4: An example showing entity, non-entity under single category and the final result of this approach after pruning out the low scorer non-entity (*courtesy: [1]*).

Facts or Assumptions:

1. This approach directly leverages the entity for task extraction, not just only the entity type information. Hence better semantics from the user queries is possible.
2. It can distinguish between tasks associated with different entities, which are under different categories.
3. It uses global picture (multiple sessions) during the procedure. Hence, this will benefit in other search related applications like, finding similar users also.

In this report, we have studied about the various paradigms to make a simple computational machine, natural or human language understandable. As we discussed in Chapter 2, Entity Recognition is mainly responsible for identifying the mentions present in the input documents and mapping them to the corresponding real world entities.

As a mention may refer to multiple real world entities, but in a particular context they will always be having just a single correct mapping of the entity. Here Word Sense Disambiguation plays the role, by removing the ambiguities using various local and global approaches, as we have discussed in Chapter 3 and 4.

After correctly disambiguating the mentions present in the input document, we need to link them to the corresponding real world entity to have more informational and intuitive web experience. Chapter 5 is lined up in the same theme. Also, there are several challenges like sparseness of knowledge base for unpopular entities, and many others. An approach is being discussed in Chapter 5 to mitigate this problem.

In the last Chapter 6, we learned about one of the importances of making natural or human language understandable to a computational machine. By extracting the entities from a log file, we can use it to predict the user's like or dislikes, behavior, needs, or anything. This information can be leveraged in many ways by any organization.

Multilingual_WSD_Report

ORIGINALITY REPORT

5 %	1 %	4 %	1 %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

- 1** Prokofyev, Roman, Gianluca Demartini, and Philippe Cudré-Mauroux. "Effective named entity recognition for idiosyncratic web collections", Proceedings of the 23rd international conference on World wide web - WWW 14, 2014. **3%**
Publication
 - 2** Verma, Manisha, and Emine Yilmaz. "Entity Oriented Task Extraction from Query Logs", Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management - CIKM 14, 2014. **1%**
Publication
 - 3** www.ist.psu.edu **1%**
Internet Source
 - 4** Submitted to Deakin University **<1%**
Student Paper
 - 5** Submitted to University of South Alabama **<1%**
Student Paper
- Submitted to Majan College

EXCLUDE QUOTES ON

EXCLUDE ON
BIBLIOGRAPHY

EXCLUDE MATCHES < 15 WORDS