**Introduction to Assignments 2 & 3**

Pronominal anaphora resolution is the problem of resolving what a pronoun, or a noun phrase refers to.

Example: “**The Empress** hasn't arrived yet, but **she** should be here any minute.”

In the above example, the pronoun “she” is the anaphor and “The Empress” is the antecedent.

A particularly difficult case for anaphora resolution is the pronoun “*it”*, as “it” may refer to a specific noun phrase or an entire clause, as shown by the two sentences below.

1. “Now, as to this quarrel. When was the first time you heard of it?” (*Nominal anaphoric*)
2. “You have been with your mistress many years, is it not so?” (*Clause anaphoric*)

# Incorrect classification of the pronoun “it” is an important culprit behind the failure of question answering systems, particularly those that handle social-network data. This is an important issue for software engineering research too, for example, in the context of mining software-related information from Twitter. It is therefore important to develop automated approaches that can accurately classify the occurrences of “it” in a given text.

# **Assignment 2: Working with the NLP Pipeline (deadline: Nov. 8)**

This assignment is about extracting information that will later (in Assignment 3) be used for classifying the occurrences of “it”into *Nominal anaphoric* and *Clause anaphoric*.

We use a dataset that contains **513** sentences where each occurrence of “it” has been classified into either NOMINAL or CLAUSE. The file is in tsv format, including the sentence and the anaphoric class separated by a tab. In Assignment 2, we are not concerned with the class; our focus will be on extracting information (features) from the sentences.

## Features to Extract

For each occurrence of “it”, extract the following features for the encompassing sentence:

|  |  |  |  |
| --- | --- | --- | --- |
| # | Feature Description | Type | NLP Technology Required |
| F1 | The position of *“it”* in the sentence considering the number of tokens, e.g., “*it”* is the first token in the sentence “It was raining!”, so the value of this feature for our example sentence is 1. | Numeric | Tokenization |
| F2 | The length of the sentence in terms of tokens, e.g., “It was raining!” has four tokens, so the value is 4. | Numeric | Tokenization |
| F3 | The number of punctuations, e.g., “It was raining!” has one punctuation mark, so the value is 1. | Numeric | Tokenization + [optional] POS tagging (hint: instead of defining a set of punctuations, POS-tagger has a value “punctuation” which can be used to count punctuations in a sentence) |
| F4 | The number of preceding noun phrases (that is: how many noun phrases come before a certain instance of “it”?) For F4, we count only the atomic noun phrases. | Numeric | Tokenization, POS tagging, chunking (or constituency parsing) |
| F5 | The number of noun phrases that follow a given instance of “it”. Like in F4, F5 is concerned with atomic noun phrases. | Numeric | Tokenization, POS tagging, chunking (or constituency parsing) |
| F6 | Does the pronoun “*it”* immediately follow a prepositional phrase? (Yes/No) | Boolean | Tokenization, POS tagging, chunking (or constituency parsing) |
| F7 | The part-of-speech (POS) tags of the four tokens immediately preceding and the four tokens immediately succeeding a given instance of “it”. If there are less than four tokens before/after, assign a special value, e.g., ABS (absent) to the missing POS tags. | Nominal (including an enumeration of the POS tags for this feature) | Tokenization, POS tagging |
| F8 | Is the occurrence of “it” followed by an -ing form of a verb? (Yes/No) | Boolean | Tokenization, POS tagging |
| F9 | Is the occurrence of “it” followed by a preposition? (Yes/No) | Boolean | Tokenization, POS tagging |
| F10 | The number of adjectives that follow the occurrence of “it” in the sentence. | Numeric | Tokenization, POS tagging |
| F11 | Is the pronoun “it” preceded by a verb? (Yes/No) | Boolean | Tokenization |
| F12 | Is the pronoun “it” followed by a verb? (Yes/No) | Boolean | Tokenization |
| F13 | Is the pronoun “it” followed by an adjective? (Yes/No) | Boolean | Tokenization |
| F14 | True if there is a noun phrase coming after the pronoun “it” and that noun phrase contains an adjective, otherwise false. | Boolean | Tokenization, POS tagging, chunking (or constituency parsing) |
| F15 | The number of tokens coming before the following infinitive verb (if there is one), otherwise 0. | Numeric | Tokenization, POS tagging |
| F16 | The number of tokens that appear between the pronoun “it” and the first following preposition (if there is a following preposition), otherwise 0. | Numeric | Tokenization, POS tagging |
| F17 | True if there a sequence “adjective + noun phrase” following the pronoun “*it”*, and false otherwise. | Boolean | Tokenization, POS tagging, chunking (or constituency parsing) |
| F18 | The dependency relation type with the closest word to which “*it”* is associated as a dependent. | Nominal (including an enumeration of the dependency types for this feature) | Tokenization, POS tagging, dependency parser (hint: select the relation when “it” is dependent and in case there are multiple, select the one associated with the closest word to it in the sentence) |
| F19 | True if the immediately following verb belongs to the category “weather adjectives”, and false otherwise. | Boolean | Tokenization, POS tagging, semantic analysis using wordnet (hint: check whether any sense of the verb following “it” belongs to the lexical file “verb.weather” in WordNet) |
| F20 | True if the immediately following verb belongs to the category of cognitive verbs, and false otherwise. | Boolean | Tokenization, POS tagging, semantic analysis using wordnet (hint: check whether any sense of the verb following “it” belongs to the lexical file “verb.cognition” in WordNet) |

## Deliverables for Assignment 2

1. Your implementation in the language and NLP toolkit of your choice
2. The output produced by your implementation on the given dataset (the instances of “it” in the 513 sentences provided). To facilitate using the output for Assignment 3, **you are encouraged to produce a csv file**. Each row in the csv file is the set of features extracted for a given instance of “it”. Please note that there may be multiple occurrences of “it” in one sentence. In such cases, all instances will have the same class (either NOMINAL or CLAUSE). Your implementation should support multiple instances of “it” appearing in the same sentence.
3. A report (max 3 pages) providing pseudo-code for the 20 features implemented. **The pseudo code you provide has to be readable to an outsider who may not have knowledge of the programming language and NLP toolkit you have used.**
4. An error analysis (max 1.5 pages) highlighting any mistakes made by the NLP pipeline on the sentences. For example, incorrect dependencies between “it” and other sentence parts.

# **Assignment 3: The ML-based Classification (deadline Nov. 22)**

1. Turn the output from Assignment 2 into an ML feature matrix. To do so, you need to add the class, **ClauseAnaph** or **NomAnaph**, to each of the rows extracted in Assignment 2.
2. Apply the following algorithms for learning: *Decision Tree, Random Forest, Logistic Regression and SVM*. Use the evaluation metric **accuracy** alongside **ten-fold cross validation** to select the best-performing classifier for the pronominal anaphora resolution of the pronoun “it” (hint: decision tree is J48, and SVM is libSVM in WEKA). Describe your accuracy results.
3. Use SMOTE on the dataset to make it more balanced (hint: SMOTE can be found in WEKA, Filter 🡪 supervised 🡪 instance 🡪 SMOTE. Rerun the algorithms from (2)). Describe your accuracy results.
4. Optimize the hyper-parameters of the above algorithms using CVParameterSelection. The optimization metric should be accuracy (hint: in WEKA you find this under classifier 🡪 Meta 🡪 CVParameterSelection).

Specifically, optimize:

* 1. C and M for Decision Tree, C in range [0.1 – 0.5] with 5 steps, M in range [2.0 – 10.0] with 9 steps.
  2. I and K for Random Forest, I in range [100,1000] with 10 steps, K in range [5, 50] with 10 steps.
  3. Ridge in logistic regression, ridge in range [1.0E-10, 1.0E-1] with 10 steps.
  4. Cost and gamma for SVM, both cost and gamma in range [1.0E-3, 1.0E3] with 7 steps.

1. Rerun the algorithms with optimal parameters. Describe your accuracy results.

Use information gain to select the most influential features. (hint: information gain is found in WEKA under attribute selection🡪Attribute Evaluator: InfoGainAttributeEval; the ranker will be selected by WEKA automatically). Provide the ranking of the features.

## Deliverables for Assignment 3

1. Your feature matrix
2. A report (max 3 pages), discussing the following: (1) your accuracy analysis in the different steps, (2) your feature ranking, and (3) lessons learned.