1. dataCategoris

* description: a stack used to record current data’s category.
* data type: [] with string elements, element’s value could be: webtop-g, pron, sn-g, gram-g, x-g, use, collapse, idm-gs.
* 注1: 这里没有cf, exp, pos因为” cf, exp, pos”的需要通过更上级的标签(sn-g 或 x-g)来判断其数据的类别.
* 注2: 这里没有def标签, 因为某些单词本来就没有def 标签(如blazon, 只有cf 和 exp标签), 所以我们只能通过sn-g来判断当前data是不是definition的一部分.
* init value: []

1. definitionTuple

* description: a group of string member. When necessary, we can use all of these members to structure a single definition and relative example string.
* data type: {“definition”: {“def”: ““, “cfexp”: ““, “gram”: ““, “pos”: ““, “use”: ““}, “examples”: [example string 1, example string 2, …]}
* init value: {“definition”: {“def”: ““, “cfexp”: ““, “gram”: ““, “pos”: ““, “use”: ““}, “examples”: []}
* algorithm to add definitionTuple into definitionTuples:

AdddefinitionTupleIntoDefinitionTuples():

string = definitionTuple[“definition”][“gram”] + definitionTuple[“definition”][“pos”]

string = string + definitionTuple[“definition”][“use”]

string = string + definitionTuple[“definition”][“cfexp”]

string = string + definitionTuple[“definition”][“def”]

definitionTuples.append({“definition”: string, “examples”: definitionTuple[“examples”]})

* example for definitionTuple[“definition”][“cfexp”]: the “Linking Verb” in 1st and 7th definition of “look”
* example for definitionTuple[“definition”][“pos”]: the “Linking Verb” in 4th definition of “look”
* example for definitionTuple[“definition”][“gram]”: the “intransitive, transitive” in 3rd definition of “look”, the “usually passive” in definition of blazon
* example for definitionTuple[“definition”][“use”]: the “not usually used in the progressive tenses” in 5th definition of “look”

1. definitionTuples

* description: a list including all of definitionTuple.
* data type: [{“definitions”: ““, “examples”: [“example string 1”, “example string 2”, …]}]

1. example

* description: a group of string member. When necessary, we can use all of these members to structure a single example string.
* data type: {“x”: ““, “cfexp”: ““}
* init value: {“x”: ““, “cfexp”: ““}
* algorithm to add example into definitionTuple:

AddExampleIntoDefinitionTuple():

string = example[“cfexp”] + example[“x”]

definitionTuple[“examples”].append(string)

1. wordName

* description: the word we are studing
* data type: sting
* init value: ““

1. wordClass

* data type: string
* element value range: noun, verb, adjective

1. wordPhonType

* description: There 2 “<span class= “phone”> </span>“ area relative to phonetic symbol. We only need “NAmE” phonetic symbol. In order to discriminate these 2 types of phonetic symbol, we defined the “wordPhonType”.
* Event ---- action

Wherever the </span> for <span class= “phone”>, set wordPhonType = ““.

When <span class=“name”> and dataCategoris.top() == “ phon”, set wordPhonType = data between <span class=“name”> and <span>

1. <span class= “wrap”>, <span class= “sep”>, <span class= “cl”>, <span class= “ptl”>, <span class= “prefix”>, <span class= “gl”>, <span class= “reg”>, <span class= “xh”>, <span class= “xs”>

* Example for <span class= “wrap”> <span class= “ptl”>: the area between <span class= “phon”> and </span> of “look”
* Example for <span class= “xh”> <span class= “xs”>: the 2nd definition of “blazon”
* Example for <span class= “prefix”>, <span class= “gl”>: the 5th definition of “look”
* Example for <span class= “reg”>: the 6th definition of “look”

|  |  |  |
| --- | --- | --- |
| event | pre-condition | action |
| <div, class, “webtop-g”> | dataCategoris.top() not in [“collapse”, “idm-gs”, “un”] | 1. dataCategoris.append(“webtop-g”) |
| </div, class, “webtop-g”> | ---- | 1. dataCategoris.pop() |
| <span class=“sn-g”>  or  <li class=“sn-g”> | dataCategoris.top() not in [“collapse”, “idm-gs”, “un”] | 1. dataCategoris.append(“sn-g”) |
| </span class=“sn-g”>  or  </li class=“sn-g”> | ---- | 1. dataCategoris.pop() 2. process definitionTuple:    1. AddDefinitionTupleIntoDefinitionTuples()    2. reinitialize definitionTuple |
| <div, class, “phon”> | dataCategoris.top() not in [“collapse”, “idm-gs”, “un”] | 1. dataCategoris.append(“phon”) |
| </div, class, “phon”> | ---- | 1. wordPhonType = ““ 2. dataCategoris.pop() |
| <span class=“x-g”> | dataCategoris.top() not in [“collapse”, “idm-gs”, “un”] | 1. dataCategoris.append(“x-g”) |
| </span class=“x-g”> | ---- | 1. If example is not empty:    1. AddExampleIntoDefinitionTuple()    2. reinitialize example 2. dataCategoris.pop() |
| <div, class, “use”> | dataCategoris.top() not in [“collapse”, “idm-gs”, “un”] | 1. dataCategoris.append(“use”) |
| </div, class, “use”> | ---- | 1. dataCategoris.pop() |
| <span class=“gram-g”> | dataCategoris.top() not in [“collapse”, “idm-gs”, “un”] | 1. dataCategoris.pop() |
| </span class=“gram-g”> | ---- | 1. dataCategoris.pop() |
| </span class=AttrValue>,  the AttrValue in [“collapse”, “idm-gs”, “un”] | ---- | 1. dataCategoris.append(AttrValue) |
| </span class= AttrValue>,  the AttrValue in [“collapse”, “idm-gs”, “un”] | ---- | 1. dataCategoris.pop() |
| data in  <span class= AttrValue> and </span>,  the AttrValue in [“wrap”, “sep”, “cl”, “ptl”, “prefix”, “gl”, “reg”, “xh”, “xs”]  or  <strong class=“pseudo”> and </span> | dataCategoris.top() == “sn-g” | 1. definitionTuple[“definition”][“def”] += data |
| dataCategoris.top() == “x-g” | 1. example[“x”] += data |
| dataCategoris.top() == “gram-g” | 1. definitionTuple[“definition”][“gram”] += data |
| dataCategoris.top() == “use” | 1. definitionTuple[“definition”][“use”] += data |
| dataCategoris.top() == “phon” and wordPhonType = “NAmE” | 1. wordPhon += data |
| data in  <span class=“pos”> and </span> | dataCategoris.top() == “sn-g” | 1. definitionTuple[“definition”][“pos”] += data |
| dataCategoris.top() == “webtop-g” | 1. wordClass = data |
| data in  <span class=“def”> and </span> | dataCategoris.top() == “sn-g” | 1. definitionTuple[“definition”][“def”] += data |
| data in  <span class=“x”> and </span> | dataCategoris.top() == “x-g” | 1. example[“x”] += data |
| data in  <span class= AttrValue> and </span>,  the AttrValue in [“cf”, “exp”] | dataCategoris.top() == “sn-g” | 1. definitionTuple[“definition”][“cfexp”] += data |
| dataCategoris.top() == “x-g” | 1. example[“cfexp”] += data |
| data in  <span class=“gram”> and </span> | dataCategoris.top() == “gram-g” | 1. definitionTuple[“definition”][“gram”] += data |
| data in  <span class=“use”> and </span> | dataCategoris.top() == “use” | 1. definitionTuple[“definition”][“use”] += data |
| data in  <h2 class=“h”> and </h2> | dataCategoris.top() == “webtop-g” | 1. wordName = data |
| data in  <span class=“name”> and </span> | dataCategoris.top() == “phon” | 1. wordPhonType = data |