classifier

Classifier

```
Classifier(self, model=None)
```

A classifier based on calculation of distances between words. It can classify the text in forbidden/restrict (ie. 'armas', 'cigarro', 'prostituição', 'remédios', serviços) or permitted classes.

Attributes:

model (KeyedVector object, None): a word2vec trained from wikipedia(portuguese) model. Otherwise, if it is 'None', the model will be trained in the initialization of the classifier. Default value: None status (str): indicates the text's level of processinf during classification.

calc_dists

```
Classifier.calc_dists(self, word, kws)
```

Calculates the distance between a word and a list of words based on the similarity of their word2vec representation.

Input:

```
- word (str): The word that will be compared (based on similarity) to a list of
keywords.
- kws (list): List of Keywords.
```

Output:

```
- dists (list): a list of distances (float) calculated between the word and each
keyword in kws.
```

check_in_vocab

```
Classifier.check_in_vocab(self, word)
```

Checks whether a word is in the vocabulary of the model or not.

Input:

```
- word (str): the word to be verified.
```

Output:

```
- boolean valeu indicating if the word is part of the model's vocabulary.
```

rm_unseen

```
Classifier.rm_unseen(self, words)
```

Given a list of words, returns a list of those that are part of the model's vocabulary.

Input:

```
- words (list): A list of strings.
```

Output:

```
- list of strings of word that are present in the model`s vocabulary.
```

prepare_result

```
Classifier.prepare_result(self, result, url, thresh, kw_result)
```

Prepares the answer structure to be displayed in the website for the user.

Input:

```
- result (dict): maps from label to veredict.
- url (str): an url string.
- thresh (float): minimum similarity for a keyword to be considered present in the
content.
- kw_result (dict): maps from label name to a pandas` series which maps from
keywords to similarity
```

Output:

```
- answer (dict): contains url, the classification, the reasons (keywords and
label).
```

classify

```
Classifier.classify(self, url, kws, labels, dist_thresh=0.2, kws_thresh=0.49)
```

Classifies an url based on the word2vec similarity of words extracted from its html content. The result is the output of the prepare_result function.

Input:

```
- url (str): an url string.
- kws (list): a list of Keywords from Keyword database.
- labels (list): a list od Labels from Label database.
```

Optional: - dist_thresh (float, 0.20): minimum similarity for a label to be considered present in the content. - kws_thresh (float, 0.49): minimum similarity for a keyword to be considered present in the content.