

HOUSING INITIATIVES:

Its Relevancy and Inclusion in Social Protection Programs –
Documentation and Results

CONCORD

Professor Tanu Kumar | Garrett Stephens

Global Research Institute at the College of William and Mary
427 Scotland St, Williamsburg, VA 23185

Purpose of Analysis

The purpose of this analysis is to scrutinize the relevance and inclusion of housing initiatives in social protection programs and systems. The World Bank defines social protection systems as initiatives to “help the poor and vulnerable cope with crises and shocks, find jobs, invest in the health and education of their children, and protect the aging population.” (*The World Bank in Social Protection*, World Bank) Besides the general definition, social protection systems can be broken down into different categories – the Governance and Social Development Resources Centre (GSDRC) divides social protection into the following groups: Social Assistance, Social Care, Social Insurance, and Labor Market Policies and Interventions. (*Types of Social Protection*, GSDRC) These initiatives attempt to cover a span of needs – direct cash, fuel and energy, health insurance, job security and retention, housing assistance, etc. Housing initiatives aim to “prevent homelessness and facilitate access to adequate housing.” (ILO, 2019) This analysis uses an aggregation of academic papers, books, briefs, reports, and websites to illustrate how prevalent housing policies are relative to their other program counterparts. This analysis ultimately uses the NLTK and FuzzyWuzzy python library to count the number of key phrase mentions via a python script.

Versions and Iterations

Rudimentary Versions

Version 1 and 2 of the analysis consists of aggregating valid sources from pdf files into text files which is legible by python. A simple “social protection programs” google search was conducted to find these sources in attempts to see how prevalent housing-centered programs are as talking points in these reports and papers. The rudimentary python script output was the most common words by mention count on a line graph – common English words like “the,” “and,” “then” along with the names of the agencies were taken out (known as stopwords in the NLTK python library).

Version 3 finds the number of times a word was mentioned, and each word has been categorized based on word association – e.g., the words ‘adequate,’ ‘affordable,’ and ‘homelessness’ are associated with housing. The issue with this approach was that not every mention of the word ‘affordable’ is indeed in reference to a housing program – e.g., ‘affordable’ can also be discussing the cost of living, access to education, and the cost of food. This approach

provided us the common words found from the entire source list but lacked a valid conclusion regarding housing initiatives. Version 4 decreased the number of categories while increasing the number of sources, but the methodology was the same as V3. V4 also initiated and completed a qualitative analysis of each source via a description. Phrases for both versions were found by making each phrase string as one string via an underscore – e.g., **affordable housing** becomes **affordable_housing**.

Version 5 grouped the social protection phrases in the following categories: transfers, subsidies, poverty, labor market, food, health protection, insurance, children, and other. The words ‘credit’ and ‘loans’ were added to the finance category (under other). Instead of associating single words with a social protection program, the script searched for hard coded phrases from the socialprotection.org glossary – many programs go by the same or similar names across agencies; therefore, searching for these common names is a more mature approach compared to loose, single word associations. The results from V5 are in figure 1.

Mature and Final Versions

Version 6 of the analysis incorporated the NLTK and FuzzyWuzzy python libraries for an upgraded approach. Not only does V6 look for common phrases that refer to social protection programs as done in V5, the NLTK’s concordance function and FuzzyWuzzy enables a hard and soft coded approach.

As mentioned previously, a hard coded word means that python is searching for a string that exactly matches the query – e.g., a query for the word “cash” will only return ‘cash’ and not ‘money,’ ‘dollar,’ or ‘dollars.’ We will call this word the ‘hard mention.’ The new addition is a ‘soft mention’ where the concordance function will gather words in the near vicinity of the hard coded word while FuzzyWuzzy determines whether these words are exact or similar matches to the phrase being search. If “cash transfer” is the phrase being searched, then the ‘hard mention’ is “cash” while the ‘soft mention’ is “transfer.” If any of the words in the near vicinity of “cash” matches or is similar to “transfer,” then the mention will be counted – this enables us to detect not only “cash transfer,” but also “transfer of cash,” “cash transfers,” and “cash and food transfers” to all mean the same social protection program. Phrases have been split into the hard and soft word based on the conventions of the social protection phrase – “cash” is the hard mention because the phrase “cash transfers” is never described as “dollar transfers” or “money

transfers” (potentially “transfer of money” is a valid mention). The soft mention is based on *flexibility* – “transfer,” “transfers,” and “transfer of” all have the same meaning. In contrast, the previous versions were incapable of capturing these like terms.

Concordance Example: Hard and Soft Mentions

This is the first result from the script when analyzing “cash transfers.”

The hard mention is “cash.” Hard mention is when the query is directly looking for every mention of the exact word “cash,” in this example highlighted in yellow – not “money” or “dollars” etc. The rest of the words near the vicinity of the hard mention is scrutinized with Fuzzywuzzy to see if the word is similar (in spelling, length, and letters, not synonyms) to the word “transfer.” Fuzzywuzzy enables us to capture similar words like “transferring, transfer, pretransfer, transferred” like in the example below in red. This mention will be counted toward the key phrase “cash transfers” because “cash” was a direct match, and the transfer match was an 80% or higher match rate according to FuzzyWuzzy.

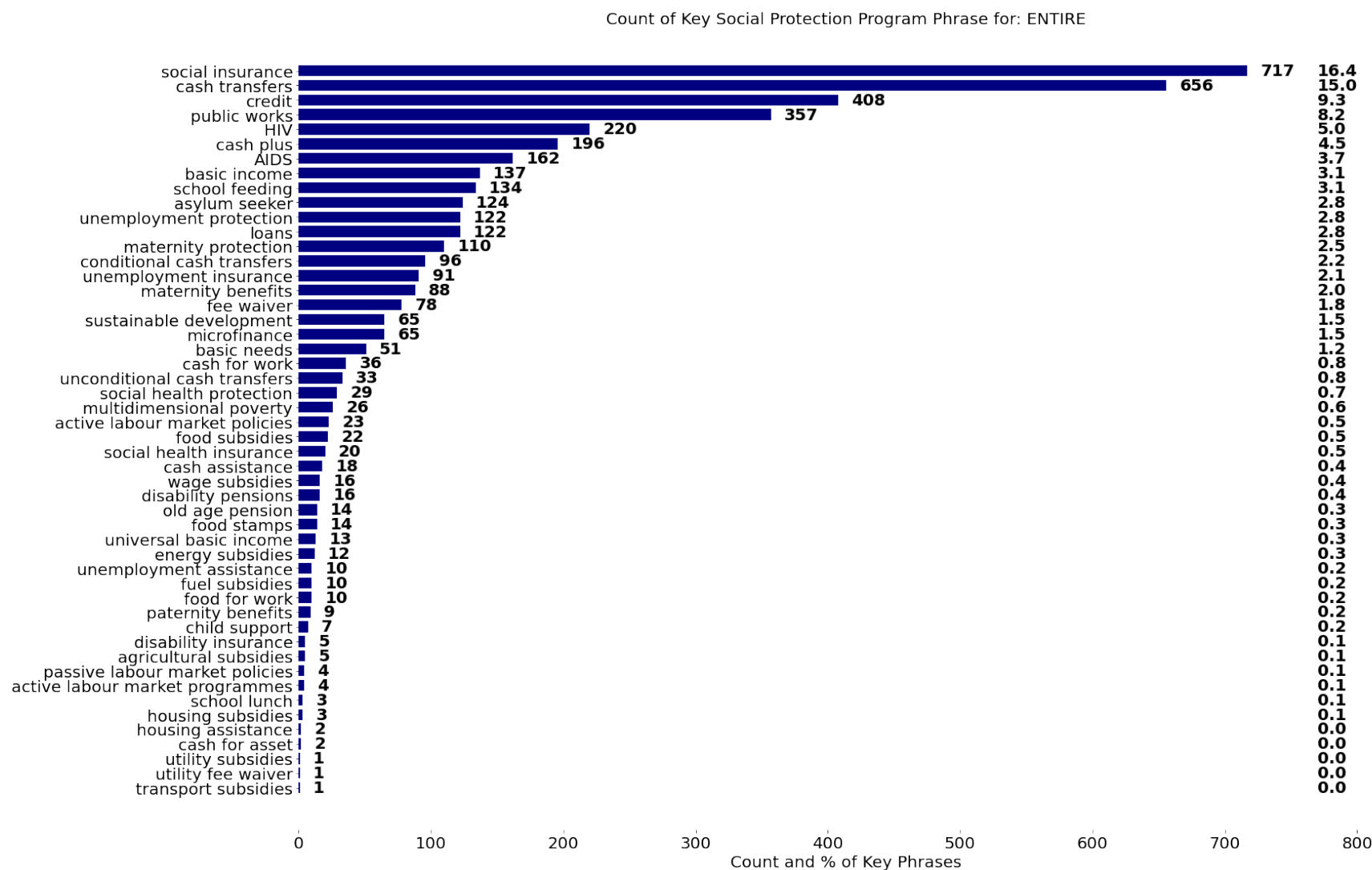
```
[ConcordanceLine(left=['prevent', '', 'deprivation', 'public',  
'works', 'projects', 'for', 'instance', 'aim', 'both', 'at',  
'transferring', 'shortterm', 'food', 'or'], query='cash',  
right=['and', 'building', 'useful', 'longterm',  
'infrastructure', 'figure', '21', 'illustrates', 'the',  
'relationship', 'between', 'these', 'measures', 'and'],  
offset=5473
```

This is an example that would not be counted because transfer is not detected:

```
ConcordanceLine(left=['million', 'people', '(', '319', 'per',  
'cent', 'of', 'the', 'population', ')', 'in', '200724', 'the',  
'latter', 'provided'], query='cash', right=['benefits', 'job',  
'training', 'and', 'small', 'loans', 'to', 'unemployed',  
'temporary', 'workers25', 'finally', 'universal', 'coverage',  
'for'], offset=94194
```

The integration of concordance was necessary because the FuzzyWuzzy will output a lower score when comparing the key phrase onto an entire sentence. The `concordance_list` command breaks down the words (i.e., tokens) into a right and left vicinity list with respect to the hard mention word which enables FuzzyWuzzy to compare the query word by each token.

Figure 1: V5 Social Protection Key Phrase Count – Hard Coded Method



This concordance and FuzzyWuzzy analysis all occurs in the `phrase_search` function. The `phrase_search` function takes 5 variables:

1. `phrases` = the list of key phrases wished to be queried
2. `data` = the list of text files desired to be scrutinized for the queried phrases
3. `category` = a one string, self-designated category name for the key phrases being searched
4. `concor_width` = the character length of the surrounding vicinity respective to the hard mention. Default value = 60.
5. `match_ratio` = how similar the scrutinized word must be to the query word conducted by FuzzyWuzzy. Default value = 80.

A Version 6.5 add-on attempts to address a potential pitfall in the hard and soft mention method. In the following made up example, the script will count this example as a mention of cash transfers because the hard mention does exist while Fuzzywuzzy detects the word “transfers” as 100% compared to the searched word “transfers.” Though I have not seen an instance like this in the actual sentences in our data, I am sure that such an instance must have occurred because of the size of the dataset. A way to increase the accuracy of the script is to decrease the width of the concordance function. This width (the variable name for width in the `phrase_search` function is `concor_width`) is the character length of each line being scrutinized; therefore, decreasing the character length of each line can increase accuracy, especially when key phrases should have both words as near ‘neighbors.’

Example sentence: ... the program supports food transfers to 3 million people. Moreover, cash grants for the elderly have been more ...

Decreasing the width to 30:

This example would not be counted toward the key phrase mention count for ‘cash transfer.’

to 3 million people. Moreover, cash grants for the elderly have b ...

The default value for the `concor_width` variable is 60 in attempts to increase the accuracy of the key phrase count and omit false counts. Moreover, the default value of the

`match_ratio` variable is 80 – this ratio seemed optimal because FuzzyWuzzy at 80 is able to detect “pretransfer” and “transferring” as the same word to “transfer” while also recognizing that “distance” and “instance” is not the same to “assistance.”

Version 7 is the final iteration known as Concor. The results from Concor are on the following pages. Each category of social protection programs and its respective key phrases are on individual horizontal bar charts. Version 8 uses the tkinter library which enables an easier user experience so that others can use this program to promote similar key phrase analysis. The interface screenshots are available after the analysis visuals.

Conclusions

The resulting visualizations from our simple google search sources indicate that housing initiatives are not discussed at great frequency compared to their social protection program counterparts because key housing phrases in total – “housing subsidies,” “housing assistance,” “housing, cash (for),” and “housing waiver” – is only 67 mentions. The initiative “cash transfers” is the most popular at 2,112 mentions. In-kind transfers and health were runner up categories behind cash transfers. Considering there are over 1 million words in our dataset, I believe seeing over 2000 mentions for cash transfers suggests that this method is robust compared to our earlier versions and illustrates a greater picture of the social protection landscape.

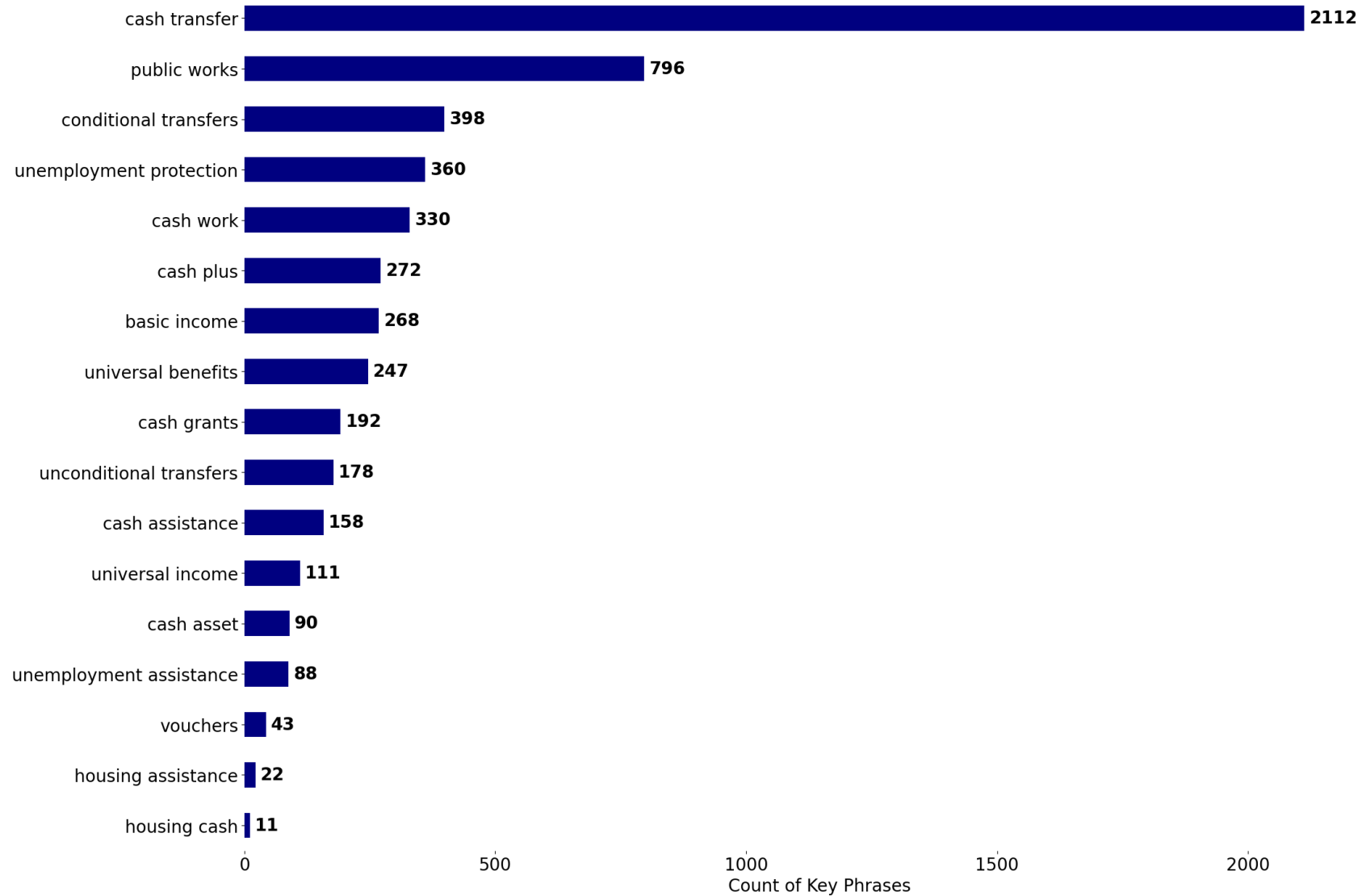
In regard to waivers, “housing waivers” is the most popular type of waiver; however, the waiver category is the smallest social protection program group at only 22 mentions total. Housing phrases are not the most mentioned in other groups like cash transfer and subsidies. While the phrase “cash transfer” has over 2000 mentions, the phrases “housing assistance” and “housing cash” only have 22 and 11 respectively. “Housing subsidies” only has 24 mentions.

One conclusion can be that housing initiatives are not a popular objective compared to other programs across agencies; however, another conclusion can be that the google search aggregation method is currently a premature representation of the social protection initiatives landscape. A simple “social protection” google search never returns reports from the UN-habitat which promotes urbanization and focuses on housing and settlements. Though, plenty of United Nations (UN) sources appear from the google query “social protection”: United Nations Children’s Fund (UNICEF), United Nations Department of Economic and Social Affairs, United

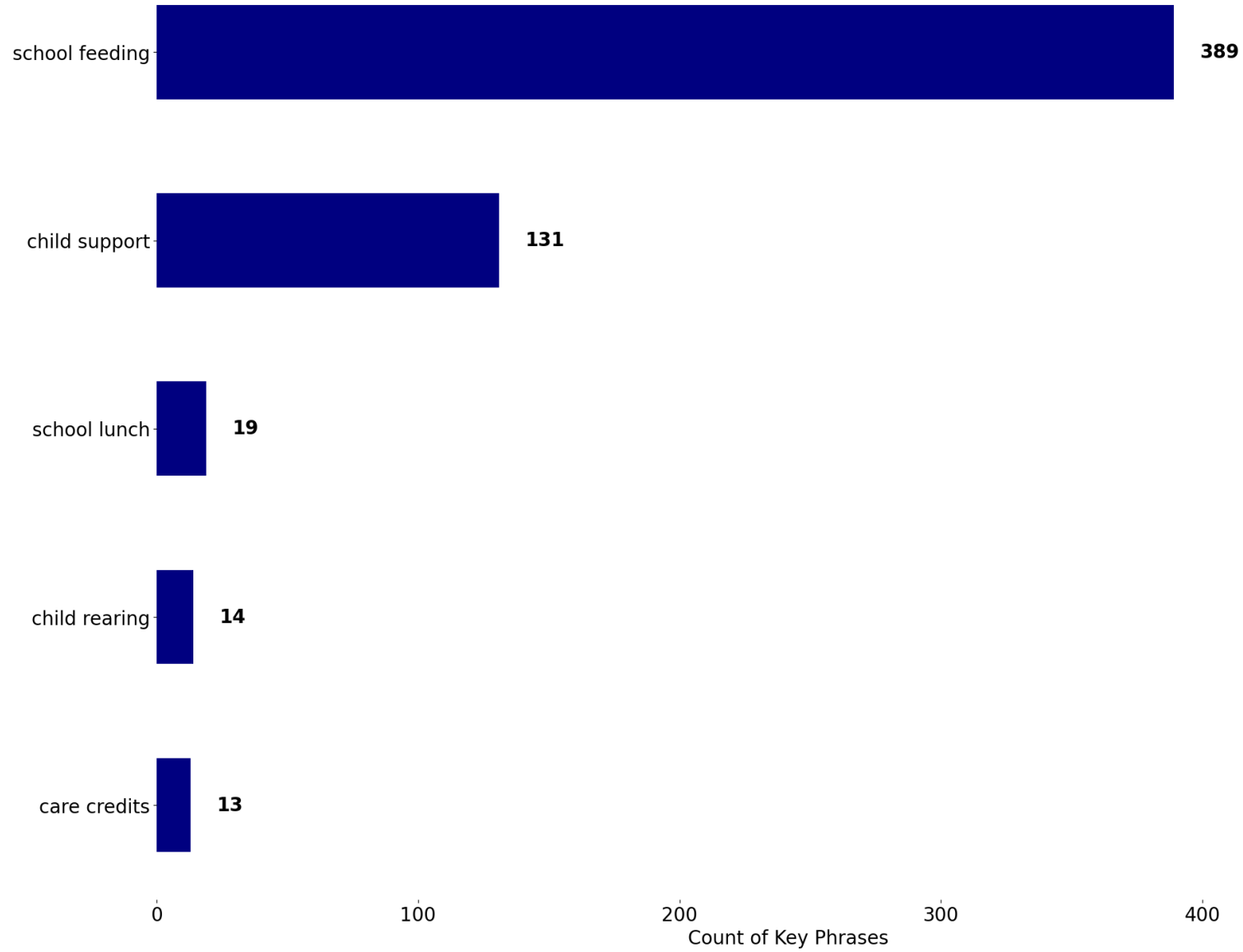
Nations High Commissioner for Refugees (UNHCR), and the United Nations Development Programme (UNDP). The google search does not actively search for housing in queries like “housing initiatives in social protection” to avoid bias.

Here are some other applications of Concor: Concor can be used to find which artist, country/minority, and forms of art are most popular for exhibits by art museum brochures, websites, and pamphlets, or a sentiment analysis to see if positive/negative words are near the vicinity of key words.

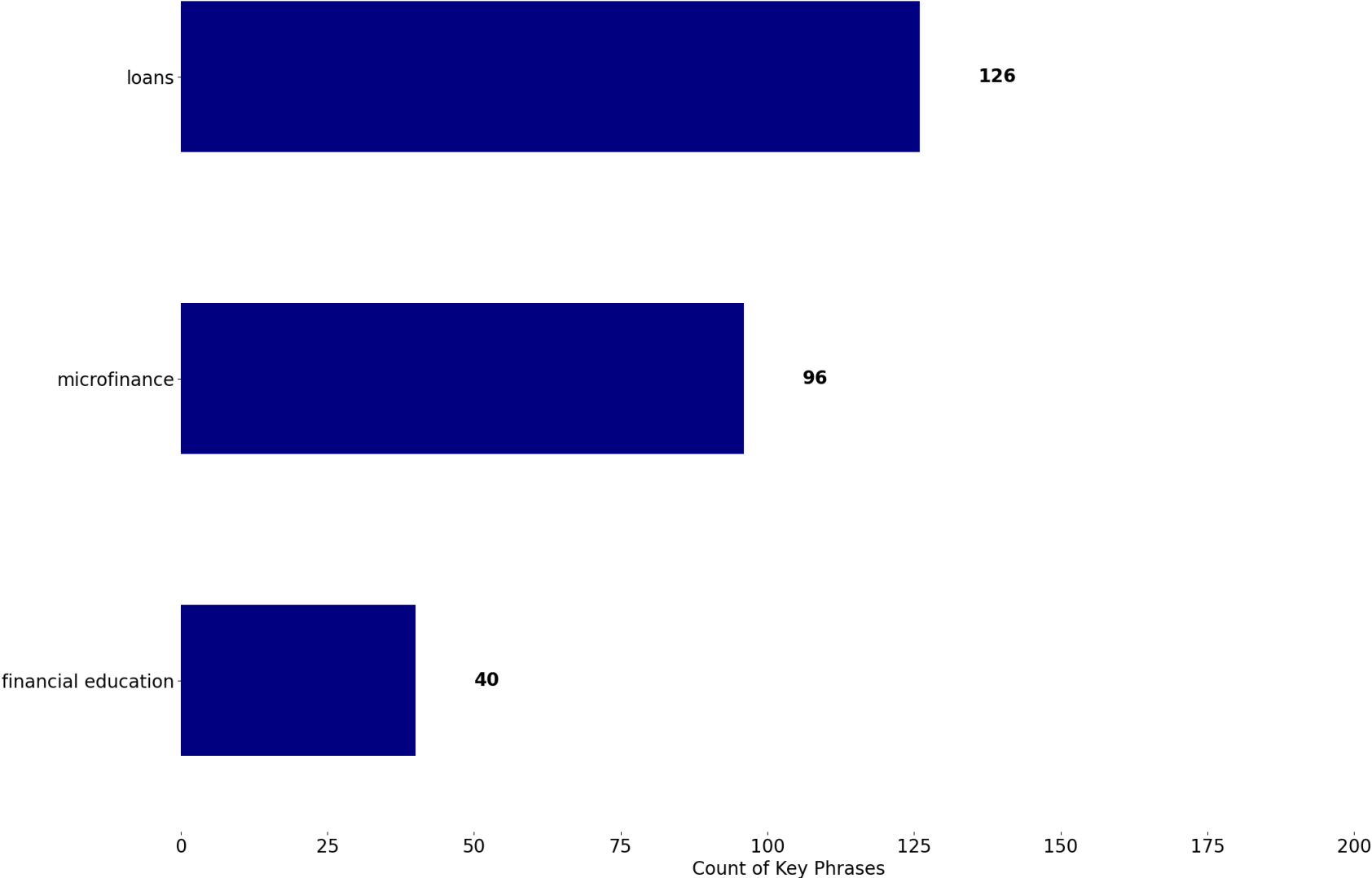
Count of Key Social Protection Program Phrases: Cash Transfer



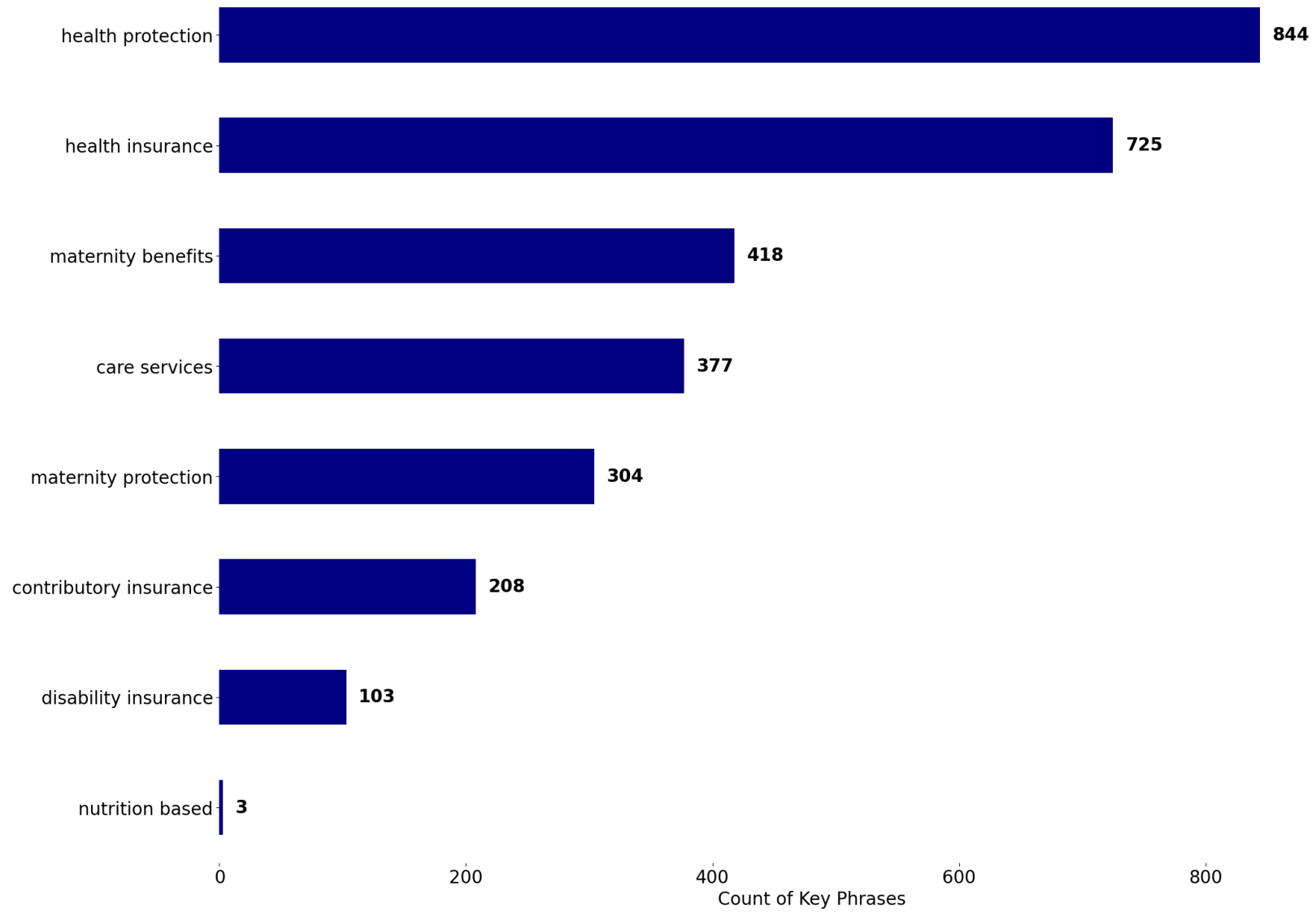
Count of Key Social Protection Program Phrases: Children



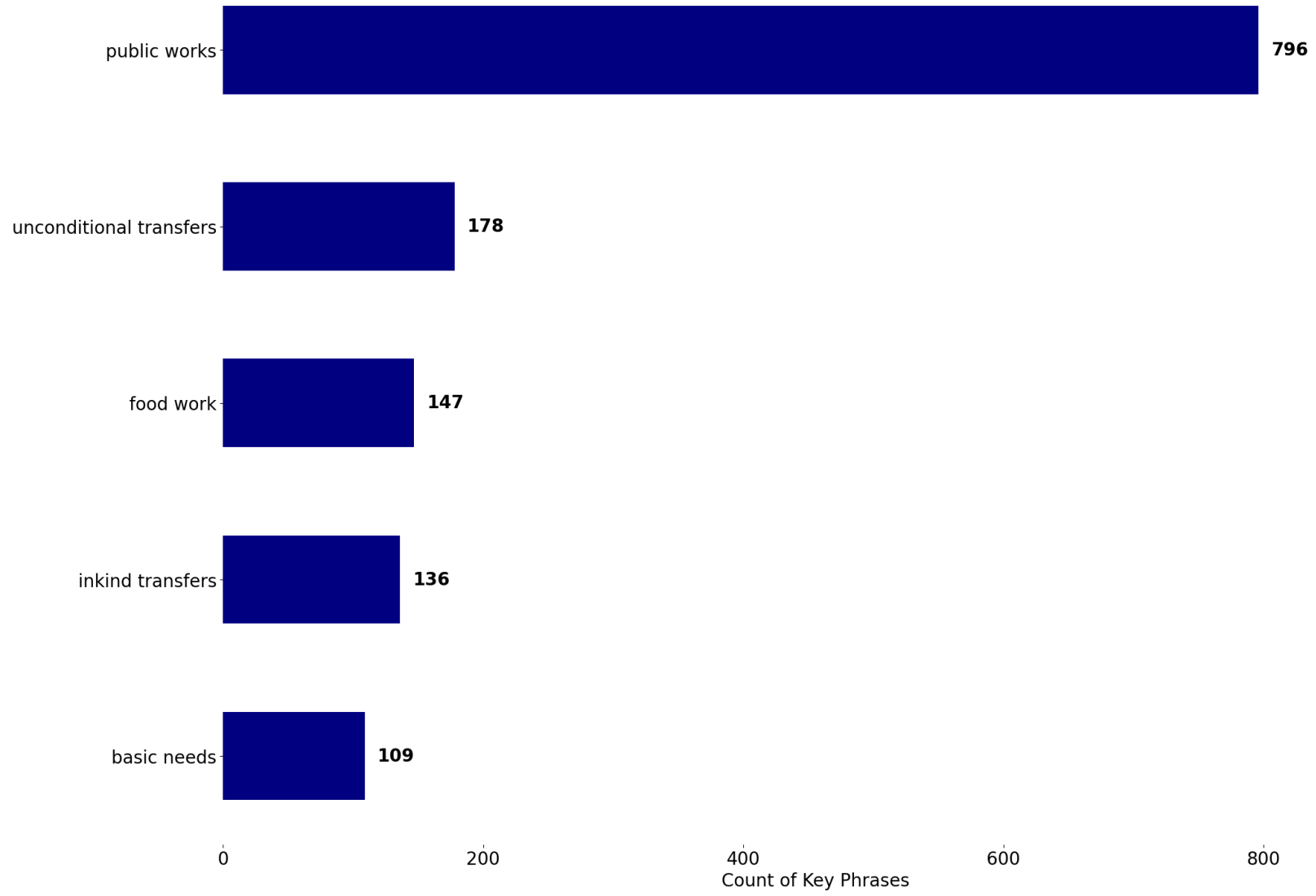
Count of Key Social Protection Program Phrases: Finance



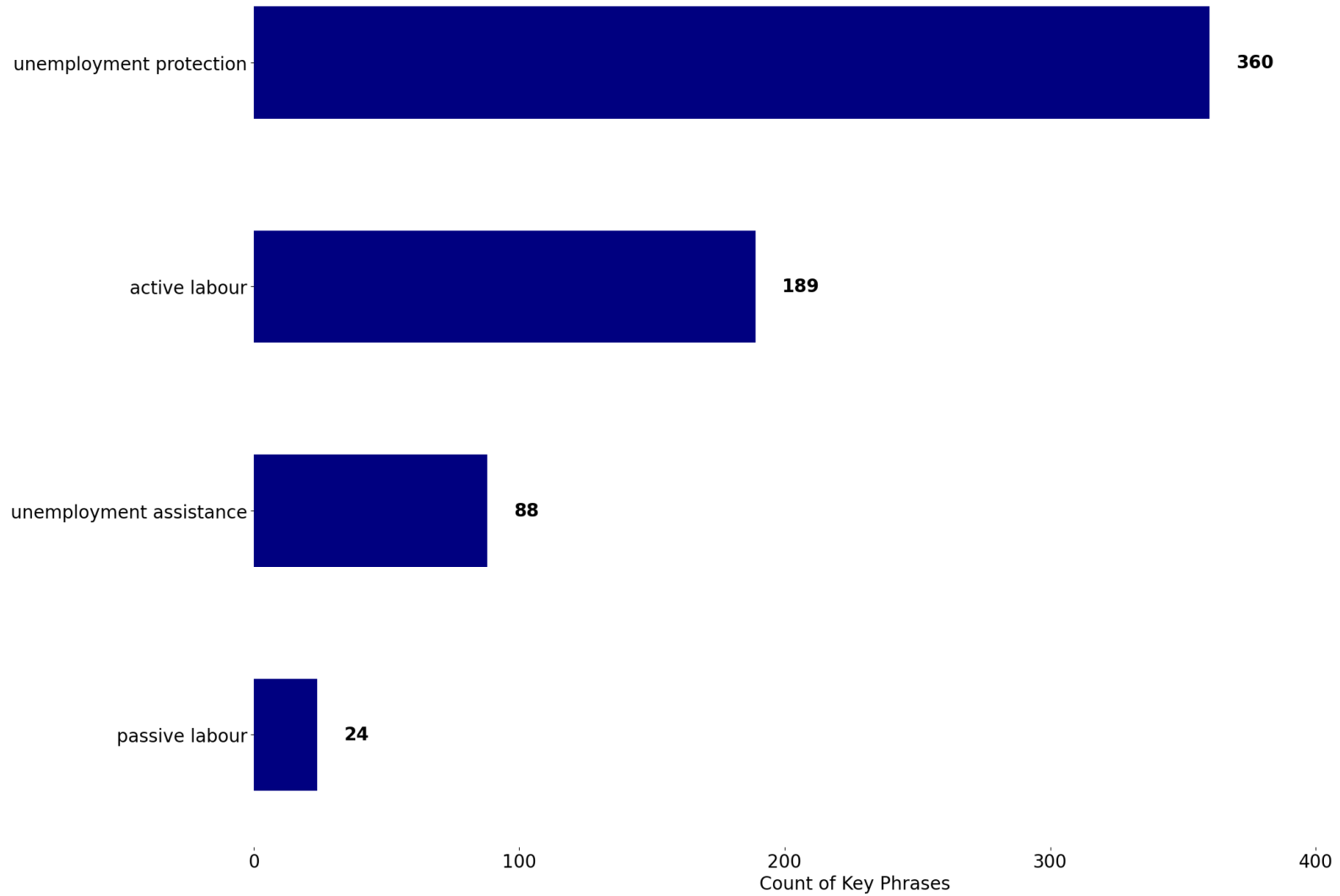
Count of Key Social Protection Program Phrases: Health



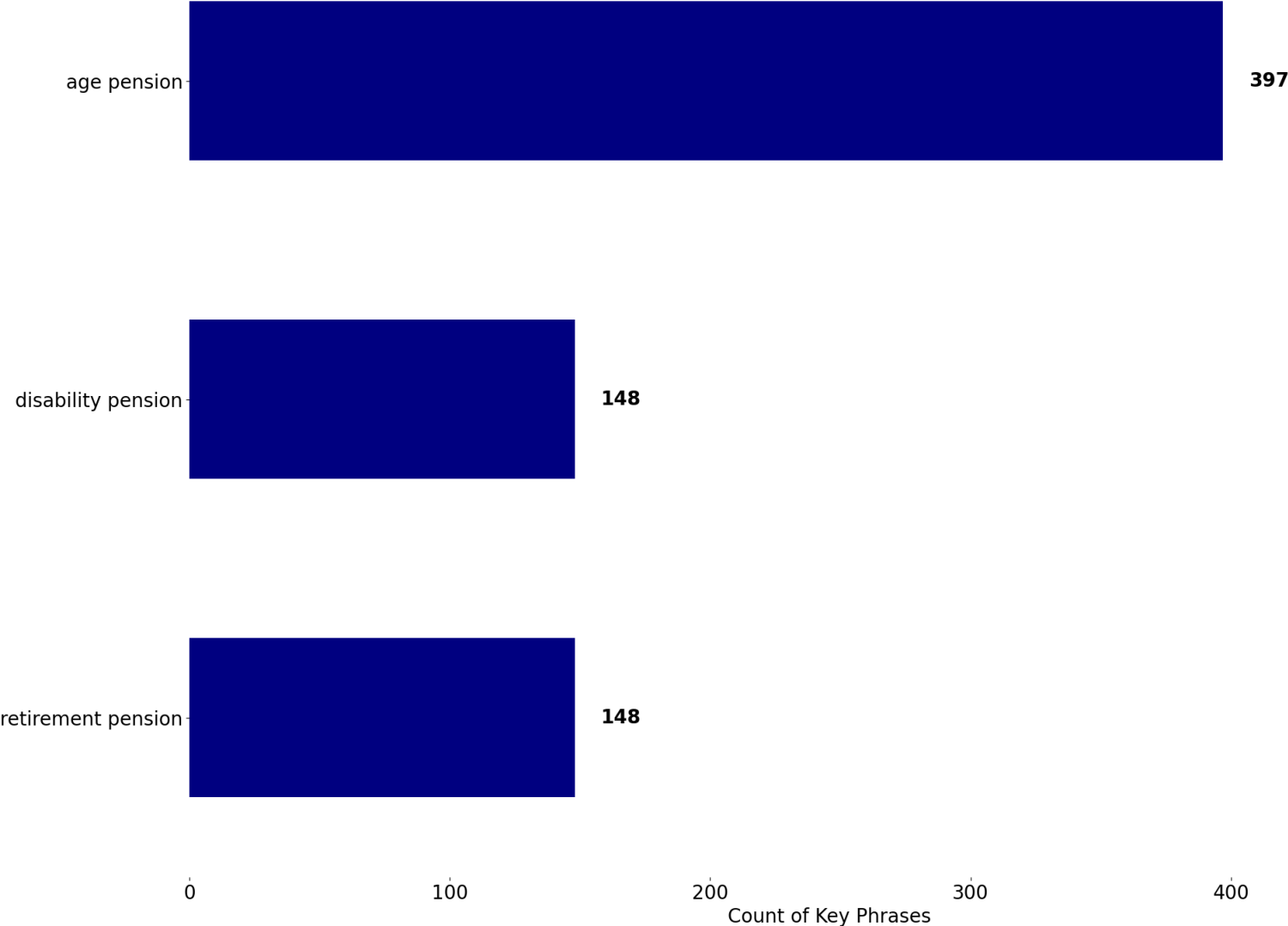
Count of Key Social Protection Program Phrases: Inkind Transfer



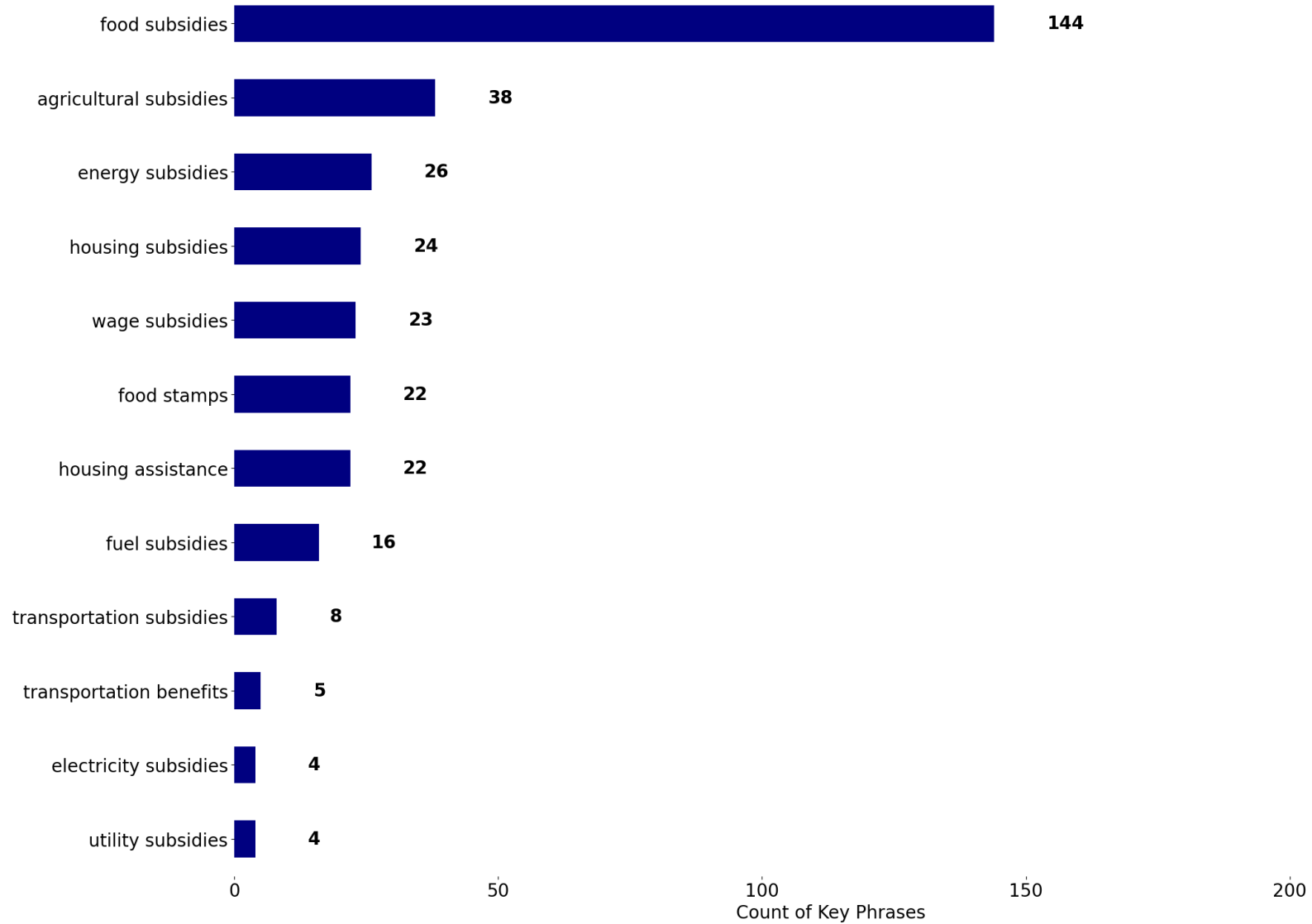
Count of Key Social Protection Program Phrases: Labor



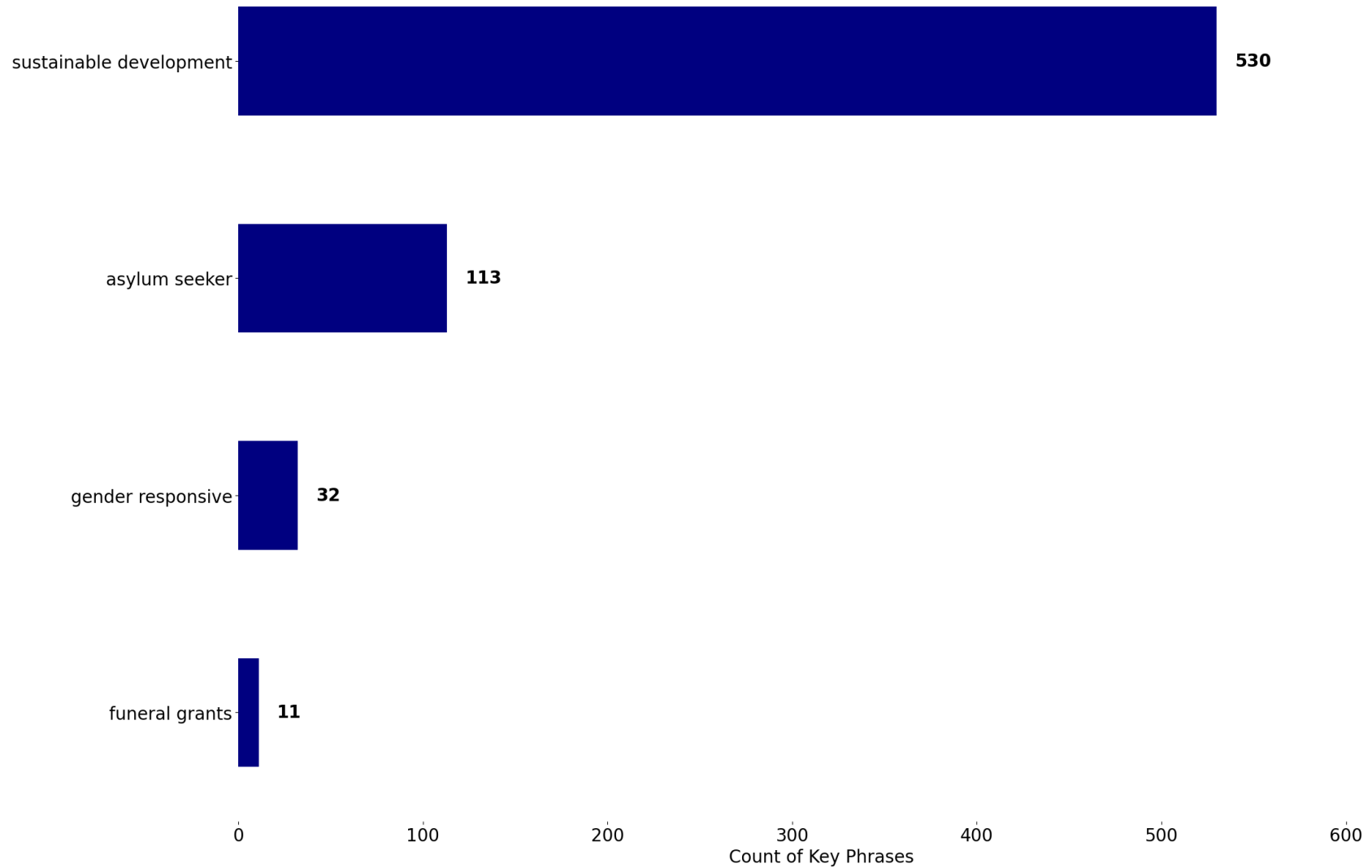
Count of Key Social Protection Program Phrases: Pension



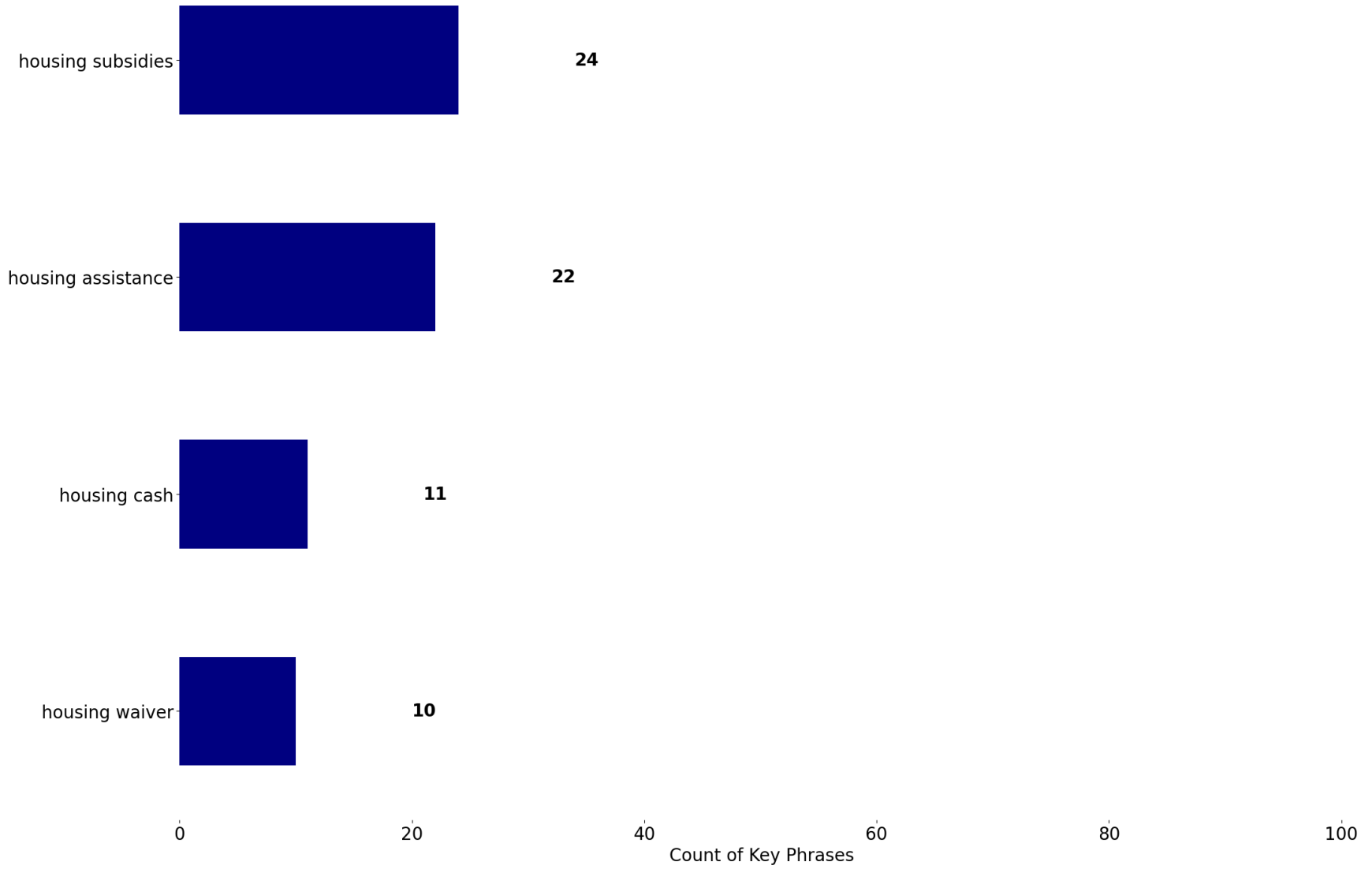
Count of Key Social Protection Program Phrases: Subsidies



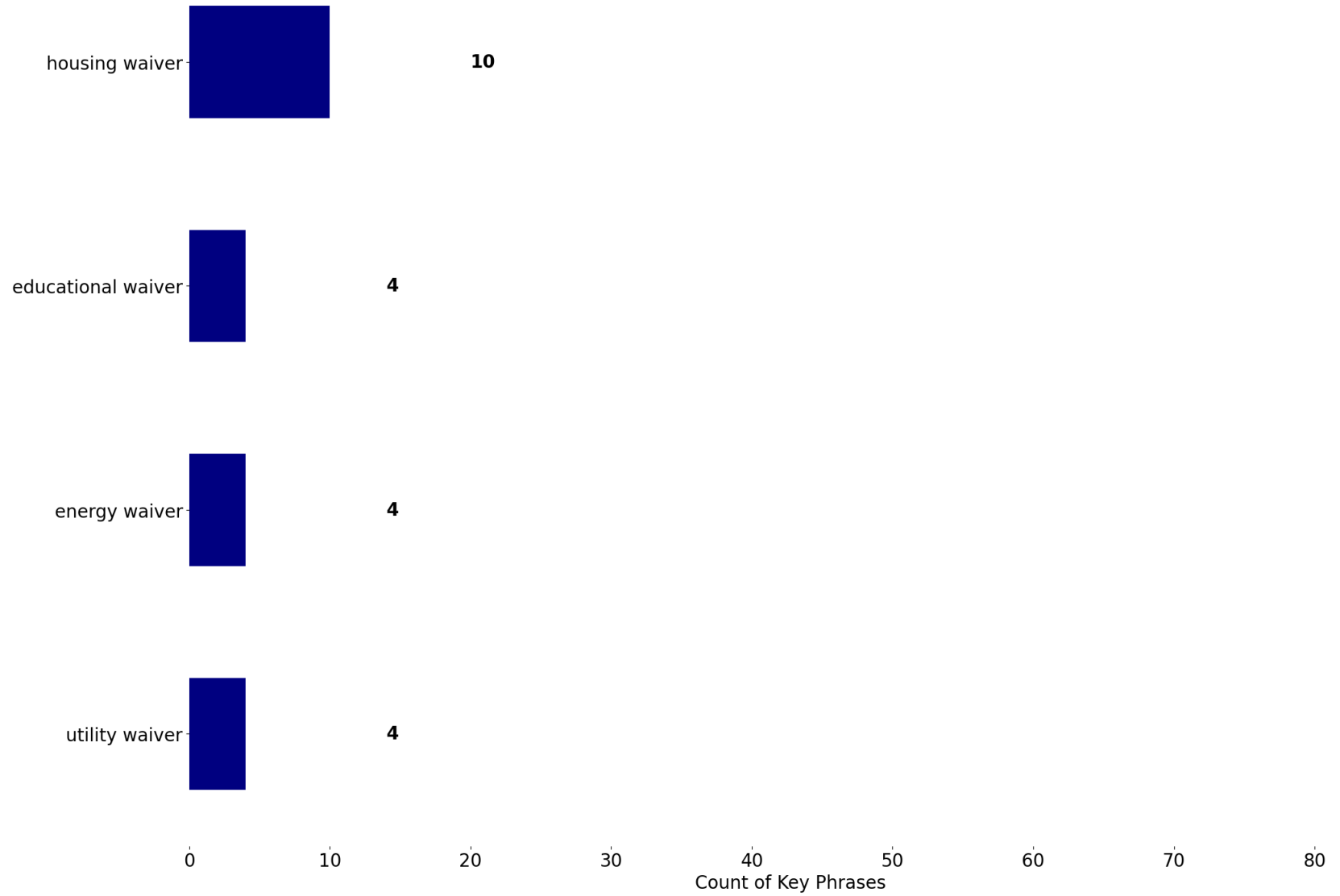
Count of Key Social Protection Program Phrases: Other



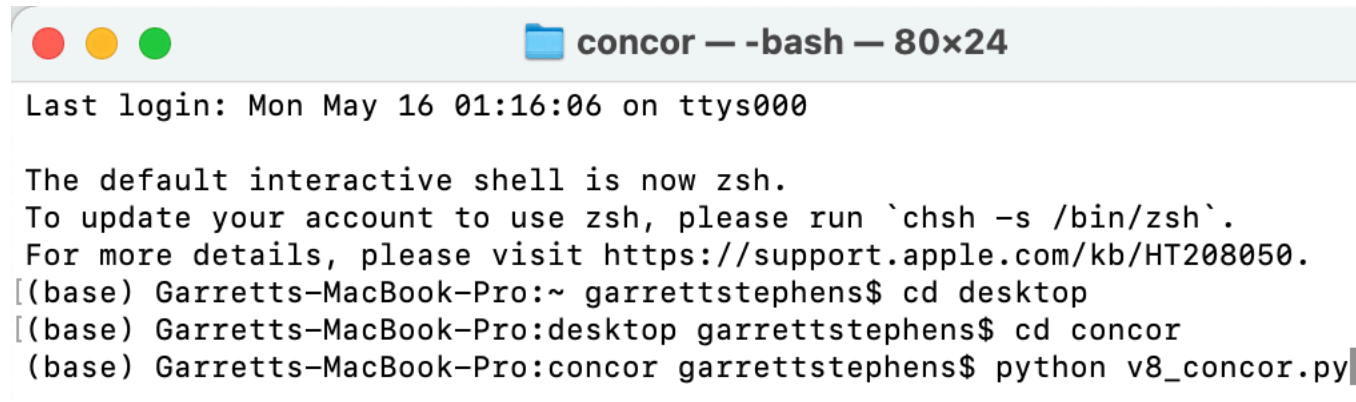
Count of Key Social Protection Program Phrases: Housing



Count of Key Social Protection Program Phrases: Waiver



On Downloading Concor File and Opening Concor:

A screenshot of a macOS Terminal window. The title bar at the top shows three colored window control buttons (red, yellow, green) on the left and a title bar with a blue folder icon and the text "concor — -bash — 80x24" on the right. The terminal content shows the last login time, a message about the default shell being zsh, instructions to update the account, and a series of terminal commands and their outputs: navigating to the desktop, then to the concor directory, and finally running a python script.

```
Last login: Mon May 16 01:16:06 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
[(base) Garretts-MacBook-Pro:~ garrettstephens$ cd desktop
[(base) Garretts-MacBook-Pro:desktop garrettstephens$ cd concor
(base) Garretts-MacBook-Pro:concor garrettstephens$ python v8_concor.py
```

On MacOS: Once you have downloaded the concor file open your Mac Terminal and direct your terminal via the cd command until you reach the concor file. If you save concor to your desktop, you can type the above commands. Typing in python v8_concor.py will open and run V8 Concor.

V8_Concor

Version 8 | 2022

*** Place all text (.txt) files to be analyzed in the directory folder: concor/data/to_read

Run Analysis

Delete Merge File

Type Desired Key Phrases to be Searched Below - Phrases can be one or two words
Example: cash transfers,loans,cash grants, ... | NO SPACES after commas

Type Desired Title for Bar Chart Below

Type Desired Graph File Name Below (Will be saved as .png file)

Type Desired Length of Soft Mention Vicinity (Recommendation: 60)
60

Type Desired FuzzyWuzzy Match Ratio Below (Recommendation: 80)
80

ERROR MESSAGES HERE:

Run Analysis: Press this button after you typed the desired key phrases to be searched, the title of the bar graph, and the file name of the bar graph png file. All files being analyzed must be text (.txt) files in the directory folder: data/to_read.

Delete Merge File: This button deletes the merge file that Concor creates after pressing the Run Analysis button. The text file that has all of your text source files in one file.

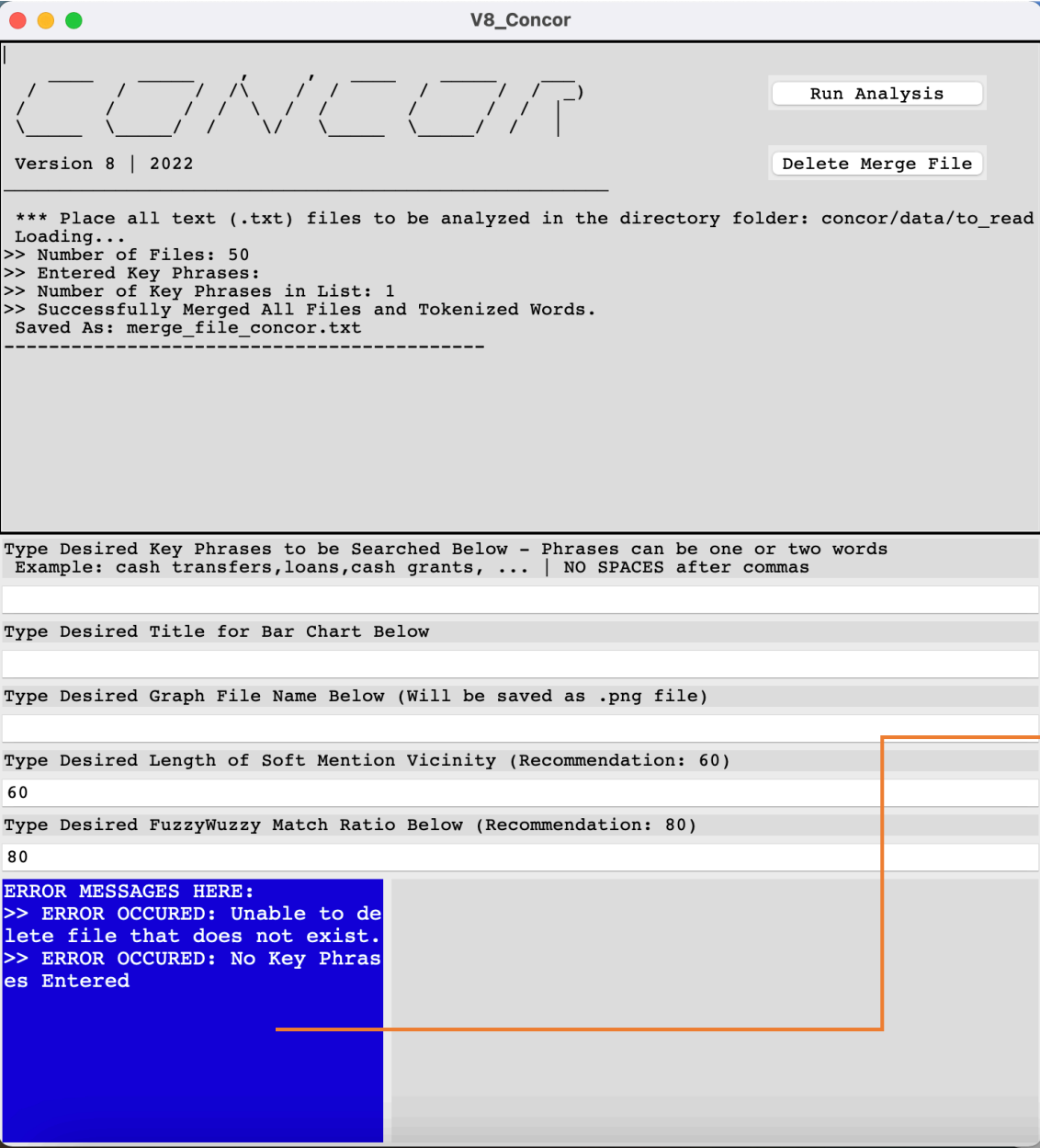
Enter Box 1: Type in your desired key phrases to be searched here. Phrases can be a key word (one word) or a key phrase (two words) – phrases cannot be three or more words. No spaces after commas! E.g.: cash transfers,loans,cash grants,housing assistance

Enter Box 2: Type in your desired title for the bar chart visualization.

Enter Box 3: Type in your desired save file name for the bar chart visualization.

Concor Width: Type desired character length of the soft mention vicinity. Default 60.

Match Ratio: Type in your desired FuzzyWuzzy Match Ratio. Default 80.



If the **Run Analysis** button is pressed before any key phrases are typed out, an Error Message will appear in the Error Message Box.

If the **Delete Merge File** button is pressed when the file does not exist, an Error will appear.

V8_Concor

Version 8 | 2022

Run Analysis

Delete Merge File

*** Place all text (.txt) files to be analyzed in the directory folder: concor/data/to_read

Loading...

>> Number of Files: 50

>> Entered Key Phrases: cash transfer,loans,public works,housing subsidies

>> Number of Key Phrases in List: 4

>> Successfully Merged All Files and Tokenized Words.

Saved As: merge_file_concor.txt

>> Successfully read through text sources

>> Total words in sources: 1726471

>> Time Lapsed to Run Analysis:0:00:02.599028

Results: Key Phrase & Mention Count:

	phrases	mentions
0	cash transfer	2112
1	loans	126
2	public works	796
3	housing subsidies	24

>> Horizontal Bar Graph Successfully Created

Saved As:test_file_may13.png

Type Desired Key Phrases to be Searched Below - Phrases can be one or two words

Example: cash transfers,loans,cash grants, ... | NO SPACES after commas

cash transfer,loans,public works,housing subsidies

Type Desired Title for Bar Chart Below

This is a Test Title

Type Desired Graph File Name Below (Will be saved as .png file)

test_file_may13

Type Desired Length of Soft Mention Vicinity (Recommendation: 60)

60

Type Desired FuzzyWuzzy Match Ratio Below (Recommendation: 80)

80

ERROR MESSAGES HERE:

>> Unique similar or exact soft mentions instances near vicinity of hard mention: housing

subsidies subsidies subsidies subsidies

subsidised subsidies

subsidised

subsidized

subsidies subsidies subsidies subsidies subsidies

subsidies subsidies

subsidies

>> Number of unique hard and soft mention combinations found: 8

The output screen displays the user inputs and analysis results. This output box can be scrolled up and down.

The number of text files, the user provided key phrases, and number of phrases is printed.

The analysis results are returned as a pandas data frame. Here we see the phrase “cash transfer” and its equivalents – e.g., “transfer of cash,” “pretransfer of cash,” etc. – total 2112 mentions. Results are ordered in mentions descending order.

The info screen displays the unique instances of soft mentions near the vicinity of the hard mention.

Here, we see the key phrase “housing subsidies” where ‘housing’ is the hard mention while the word ‘subsidies’ is the soft mention. We see instances where the word ‘housing’ was near multiple ‘subsidies.’ Perhaps this is because sources would list off many different kinds of subsidies in the same sentence.

Another soft mentions near the word ‘housing’ are ‘subsidized’ which demonstrates Concor’s ability to capture like-terms: “housing subsidies” = “subsidized housing”

```
V8_Concor

2 public works 170
3 housing subsidies 24

>> Horizontal Bar Graph Successfully Created
  Saved As:test_file_may13.png

Loading...
>> Number of Files: 50
>> Entered Key Phrases: cash transfer,loans,public works,housing subsidies
>> Number of Key Phrases in List: 4
>> Successfully Merged All Files and Tokenized Words.
  Saved As: merge_file_concor.txt
-----
>> Successfully read through text sources
>> Total words in sources: 1726471
>> Time Lapsed to Run Analysis:0:00:01.886678

Results: Key Phrase & Mention Count:
      phrases  mentions
0    cash transfer    2060
1         loans      126
2    public works     622
3 housing subsidies     16

>> Horizontal Bar Graph Successfully Created
  Saved As:test_file_may13.png

Type Desired Key Phrases to be Searched Below - Phrases can be one or two words
Example: cash transfers,loans,cash grants, ... | NO SPACES after commas

cash transfer,loans,public works,housing subsidies

Type Desired Title for Bar Chart Below

This is a Test Title

Type Desired Graph File Name Below (Will be saved as .png file)

test_file_may13

Type Desired Length of Soft Mention Vicinity (Recommendation: 60)
40

Type Desired FuzzyWuzzy Match Ratio Below (Recommendation: 80)
90

ERROR MESSAGES HERE:
>> Number of unique hard and soft mention combinations found: 5

>> Unique similar or exact soft mentions instances near vicinity of hard mention: housing
subsidies subsidies subsidies
subsidies subsidies subsidies subsidies subsidies
subsidies subsidies
subsidies

>> Number of unique hard and soft mention combinations found: 5
```

Compared to the above analysis, we have decreased the vicinity length and the match ratio to 40 and 90 respectively. Decreasing the vicinity and increasing the ratio threshold increases the accuracy of Concor but decreases the count output – a decrease in the width means we are not counting invalid appearances of the soft mention; an increase in the ratio means that FuzzyWuzzy requirement is tighter. Increasing the ratio too much can hurt our analysis e.g., the above analysis counted the word “subsidized”, but the analysis here misses this mention.

Use the info screen to find the best match ratio value for your data set.

Citation

“Social Protection.” *World Bank*, <https://www.worldbank.org/en/topic/socialprotection>.

“Social Protection Systems for All to Prevent Homelessness and Facilitate Access to Adequate Housing.” *Social Protection for All Issue Brief*, International Labour Office, Nov. 2019, <https://www.un.org/development/desa/dspd/wp-content/uploads/sites/22/2020/01/55706.pdf>.

“Types of Social Protection.” *GSDRC*, 16 Dec. 2019, <https://gsdrc.org/topic-guides/social-protection/types-of-social-protection/>.