Try one picked at random

In [85]:
```
#  Just one summarization to begin with, randomly picked ... but
#+ now with th possibility of a known seed, to allow visual
#+ comparison with after-training results.
#+ I'M NOT GOING TO USE THIS REPEATED SEED, I'm just going to
#+ use the datum at the first index to compare.

do_seed_for_repeatable = True

summarizer = pipeline('summarization',
                      model=new_model,
                      tokenizer=new_tokenizer)

if do_seed_for_repeatable:
    rand_seed_for_randrange = 137
    random.seed(rand_seed_for_randrange)
##endof:  if do_seed_for_repeatable

sample = dataset['test'][randrange(len(dataset["test"]))]
print(f"dialogue: \n{sample['dialogue']}\n---------------")

res = summarizer(sample["dialogue"])

print(f"dwb-flan-t5-small-lora-finetune summary:\n{res[0]['summary_text']}")
```

```
dialogue:
Jayden: But I don't need kids. Kids means over. At least for a woman
Brennan: Over what ?
Jayden: The end of normal life. Being pregnant, suffering because of this etc
Brennan: Hmm so I need to look for another mother to my kids then. Haha
Jayden: Being obligated to be with the. 24h. Men have only sex and they wait for kid
s while women suffer
Brennan: I don't agree...
Jayden: I wish I could do the same. Then probably i would say the same like u.
Brennan: Guys like me would be there through it all to reduce the suffering
Jayden: Physical suffering. No one can do anything with this. I wish I could just ha
ve sex and wait for a baby while having a normal life. Not getting fat, having the s
ame body, the same breast and not disgusting ... Not feeling sick, not having pain,
being able to do every day stuff even like walking...
Brennan: It's gonna happen eventually
Jayden: I was I'm a store, behind me there was a pregnant woman, she dropped some mo
ney and she couldn't even take them from the floor... I had to help her
Brennan: That's because she's about to give birth
Jayden: I hope that maybe soon they will be possible to have a child without being p
regnant. Yes! And she's suffering
Brennan: Any I'm sorry for feeding you with my bullshit
Jayden: While a man is doing his normal stuff. U mean the conversation?
Brennan: I hope you find a guy that can give you the sex you want and not get pregna
nt
Jayden: Would be awesome
Brennan: I'm gonna go to sleep now. Good night
Jayden: I said I don't want to have any children now! Maybe in the future when I hav
e a good job, I'm financially independent. Good night
---------------
dwb-flan-t5-small-lora-finetune summary:
Jayden: I said I don't want to have any children now! Maybe in the future when I hav
e a good job, I'm financially independent. Good night
```

Now, a couple summarizations with comparison to ground truth

In [86]:
```
pred_test_list = []
ref_test_list = []

sample_num = 0

this_sample = dataset['test'][sample_num]

print(f"dialogue: \n{this_sample['dialogue']}\n---------------")

ground_summary = this_sample['summary']
res = summarizer(this_sample['dialogue'])
res_summary = res[0]['summary_text']

print(f"human-genratd summary:\n{ground_summary}")
print(f"dwb-flan-t5-small-lora-finetune summary:\n{res_summary}")

ref_test_list.append(ground_summary)
pred_test_list.append(res_summary)

# deprecated, blah blah blah
rouge = load_metric('rouge', trust_remote_code=True)

#  Yes, I have just one datum, but I'm setting things up to
#+ work well with a loop (meaning lists for pred and ref).
results_test_0 = rouge.compute(
                    predictions=pred_test_list,
                    references=ref_test_list,
                    use_aggregator=False
)

# >>> print(list(results_test.keys()))
# ['rouge1', 'rouge2', 'rougeL', 'rougeLsum']
```

```
Your max_length is set to 200, but your input_length is only 133. Since this is a su
mmarization task, where outputs shorter than the input are typically wanted, you mig
ht consider decreasing max_length manually, e.g. summarizer('...', max_length=66)
```

```
dialogue:
Hannah: Hey, do you have Betty's number?
Amanda: Lemme check
Hannah: <file_gif>
Amanda: Sorry, can't find it.
Amanda: Ask Larry
Amanda: He called her last time we were at the park together
Hannah: I don't know him well
Hannah: <file_gif>
Amanda: Don't be shy, he's very nice
Hannah: If you say so..
Hannah: I'd rather you texted him
Amanda: Just text him 🙂
Hannah: Urgh.. Alright
Hannah: Bye
Amanda: Bye bye
---------------
human-genratd summary:
Hannah needs Betty's number but Amanda doesn't have it. She needs to contact Larry.
dwb-flan-t5-small-lora-finetune summary:
Amanda: Bye bye e. Alright Hannah: Hey, do you have Betty's number? Amanda: Lemme ch
eck Hannah: file_gif> Amanda: Sorry, can't find it. Amanda: Ask Larry Amanda: He cal
led her last time we were at the park together Hannah: I don'n't know him well Hanna
h: cfile_go-> Amanda: Don't be shy, he's very nice Hannah: If you say so.. Hannah: I
m'd rather you texted him Amanda: Just text him  Hannah: Urgh.. AlRIGHT Hannah: By e
Amanda: bye by e
```

In [87]:
```
print_rouge_scores(results_test_0, 0)
```

```
---------- ROUGE SCORES ----------
  --------- dialogue 1 ----------
ROUGE-1 results
[Score(
          precision=0.10465116279069768,
          recall=0.5625,
          fmeasure=0.17647058823529413)]
ROUGE-2 results
[Score(
          precision=0.023529411764705882,
          recall=0.13333333333333333,
          fmeasure=0.04)]
ROUGE-L results
[Score(
          precision=0.09302325581395349,
          recall=0.5,
          fmeasure=0.15686274509803924)]
ROUGE-Lsum results
[Score(
          precision=0.09302325581395349,
          recall=0.5,
          fmeasure=0.15686274509803924)]
```

In [ ]:
```
sample_num = 224

this_sample = dataset['test'][sample_num]

print(f"dialogue: \n{this_sample['dialogue']}\n---------------")

ground_summary = this_sample['summary']
res = summarizer(this_sample['dialogue'])
res_summary = res[0]['summary_text']

print(f"human-genratd summary:\n{ground_summary}")
print(f"dwb-flan-t5-small-lora-finetune summary:\n{res_summary}")

ref_test_list = [ground_summary]
pred_test_list = [res_summary]

rouge = load_metric('rouge', trust_remote_code=True)

results_test_224 = rouge.compute(
                    predictions=pred_test_list,
                    references=ref_test_list,
                    use_aggregator=False
)

# >>> print(list(results_test.keys()))
# ['rouge1', 'rouge2', 'rougeL', 'rougeLsum']
```

In [89]:
```
print_rouge_scores(results_test_224, 224)
```

```
---------- ROUGE SCORES ----------
  --------- dialogue 225 ----------
ROUGE-1 results
[Score(
          precision=0.17647058823529413,
          recall=0.1666666666666666,
          fmeasure=0.17142857142857143)]
ROUGE-2 results
[Score(
          precision=0.0,
          recall=0.0,
          fmeasure=0.0)]
ROUGE-L results
[Score(
          precision=0.11764705882352941,
          recall=0.1111111111111111,
          fmeasure=0.11428571428571428)]
ROUGE-Lsum results
[Score(
          precision=0.11764705882352941,
          recall=0.1111111111111111,
          fmeasure=0.11428571428571428)]
```