

Em 1]:

```
from IPython.display import Markdown , display , HTML
import warnings
warnings . filterwarnings ( 'ignorar' )
```

Análise de habilidade de diálogo para Watson Assistant

Introdução

O Dialog Skill Analysis for Watson Assistant (WA) destina-se ao uso por designers de chatbot, desenvolvedores e cientistas de dados que desejam experimentar e aprimorar seu design de habilidade de diálogo existente.

Assumimos familiaridade com o produto Watson Assistant, bem como conceitos envolvidos no design de habilidade de diálogo, como intenção, entidades, enunciados, etc.

Meio Ambiente

é necessário o python 3.6 ou superior. Para requisitos de dependência, consulte requirements.txt

Uso

1. Presume familiaridade usando um bloco de notas Python Jupyter
2. Presume um ambiente Python 3.6 ou superior
3. Instale dependências com `pip install -r requirements.txt`
4. Inicie o servidor jupyter com `jupyter notebook`
5. Selecione `skill_analysis.ipynb` para iniciar a sessão

Link para o Source Repository

<https://github.com/watson-developer-cloud/assistant-dialog-skill-analysis> (<https://github.com/watson-developer-cloud/assistant-dialog-skill-analysis>).

Em 2]:

```
! pip install --index-url https://pypi.python.org/simple -U "pip"  
! pip install --index-url https://pypi.python.org/simple -U "assistant-dialog-sk  
ill-analysis> = 1.2.1"
```

```
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secre
tstorage/dhcrypto.py:16: CryptographyDeprecationWarning: int_from_b
ytes está obsoleto, use int.from_bytes em seu lugar
  de cryptography.utils import int_from_bytes
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secre
tstorage/util.py:25: CryptographyDeprecationWarning: int_from_bytes
está obsoleto, use int.from_bytes em vez disso
  de cryptography.utils import int_from_bytes
Procurando nos índices: https://pypi.python.org/simple
Coletando pip
  Fazendo download de pip-21.2.4-py3-none-any.whl (1,6 MB)
  | ████████████████████████████████████████████████████████████████ | 1,6 MB 26,9 MB / s eta 0:
00:01
Instalando pacotes coletados: pip
  Tentando desinstalar: pip
    Instalação existente encontrada: pip 20.1.1
    Desinstalando pip-20.1.1:
      Pip-20.1.1 desinstalado com sucesso
Pip-21.2.4 instalado com sucesso
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secre
tstorage/dhcrypto.py:16: CryptographyDeprecationWarning: int_from_b
ytes está obsoleto, use int.from_bytes em seu lugar
  de cryptography.utils import int_from_bytes
/opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages/secre
tstorage/util.py:25: CryptographyDeprecationWarning: int_from_bytes
está obsoleto, use int.from_bytes em vez disso
  de cryptography.utils import int_from_bytes
Procurando nos índices: https://pypi.python.org/simple
Coletando assistente-diálogo-habilidade-análise> = 1.2.1
  Fazendo download do assistant_dialog_skill_analysis-1.2.1-py3-non
e-any.whl (51 kB)
  | ████████████████████████████████████████████████████████████████ | 51 kB 8,6 MB / s eta 0:0
0:01
Requisito já atendido: numpy> = 1.15.1 in /opt/conda/envs/Python-3.
7-OpenCE/lib/python3.7/site-packages (de assistant-dialog-skill-ana
lysis> = 1.2.1) (1.19 .2)
Requisito já atendido: pandas em /opt/conda/envs/Python-3.7-OpenCE/
lib/python3.7/site-packages (from assistant-dialog-skill-analysis>
= 1.2.1) (1.0.5)
Requisito já atendido: tabular em /opt/conda/envs/Python-3.7-OpenC
E/lib/python3.7/site-packages (de assistant-dialog-skill-analysis>
= 1.2.1) (0.8.3)
Requisito já atendido: scipy> = 1.2.0 in /opt/conda/envs/Python-3.7
-OpenCE/lib/python3.7/site-packages (de assistant-dialog-skill-anal
ysis> = 1.2.1) (1.4 .1)
Requisito já atendido: nltk em /opt/conda/envs/Python-3.7-OpenCE/li
b/python3.7/site-packages (from assistant-dialog-skill-analysis> =
1.2.1) (3.5)
Requisito já atendido: ibm-cos-sdk == 2.7.0 in /opt/conda/envs/Pyth
on-3.7-OpenCE/lib/python3.7/site-packages (de assistant-dialog-skil
l-analysis> = 1.2. 1) (2.7.0)
Coletando ibm-watson> = 4.5.0
  Fazendo download de ibm-watson-5.3.0.tar.gz (412 kB)
  | ████████████████████████████████████████████████████████████████ | 412 kB 45,4 MB / s eta 0:
00:01
  Instalando dependências de compilação ... concluído
```


ackages (from ipython> = 5.0.0-> ipykernel-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (3.0.5)

Requisito já atendido: pexpect in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (from ipython> = 5.0.0-> ipykernel-> jupyter-> assistant-dialog-skill-análise> = 1.2.1) (4.8.0)

Requisito já atendido: decorador em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de ipython> = 5.0.0-> ipykernel-> jupyter-> assistant-dialog-skill-análise> = 1.2.1) (4.4.2)

Requisito já atendido: parso <0.8.0,> = 0.7.0 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de jedi> = 0.10-> ipython> = 5.0.0-> ipykernel-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (0.7.0)

Requisito já atendido: wcwidth em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (do prompt-toolkit! = 3.0.0,! = 3.0.1, <3.1.0, > = 2.0.0-> ipython> = 5.0.0-> ipykernel-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (0.2.4)

Requisito já atendido: ipython-genutils em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (from traitlets> = 4.1.0-> ipykernel-> jupyter-> assistant-dialog-análise de habilidade> = 1.2.1) (0.2.0)

Requisito já atendido: widgetsnbextension ~ = 3.5.0 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de ipywidgets-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (3.5.1)

Requisito já atendido: nbformat> = 4.2.0 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de ipywidgets-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (5.0.7)

Requisito já atendido: jupyter-core em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de nbformat> = 4.2.0-> ipywidgets-> jupyter-> assistant-dialog-análise de habilidade> = 1.2.1) (4.6.3)

Requisito já atendido: jsonschema! = 2.5.0,> = 2.4 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de nbformat> = 4.2.0-> ipywidgets-> jupyter-> assistente-diálogo-análise de habilidade> = 1.2.1) (3.2.0)

Requisito já atendido: pyparsing> = 0.14.0 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de jsonschema! = 2.5.0,> = 2.4-> nbformat> = 4.2.0-> ipywidgets-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (0.16.0)

Requisito já atendido: attrs> = 17.4.0 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de jsonschema! = 2.5.0,> = 2.4-> nbformat> = 4.2.0-> ipywidgets-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (19.3.0)

Requisito já atendido: pyzmq> = 17 em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de notebook-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (19.0.1)

Requisito já atendido: jinja2 em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (de notebook-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (2.11.3)

Requisito já satisfeito: terminado> = 0.8.1 in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (from notebook-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (0.8.3)

Requisito já atendido: Send2Trash in /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (from notebook-> jupyter-> assistant-dialog-skill-analysis> = 1.2.1) (1.5.0)

Requisito já atendido: prometheus-client em /opt/conda/envs/Python-3.7-OpenCE/lib/python3.7/site-packages (from notebook-> jupyter-> a


```
Roda de construção para ibm-watson (PEP 517) ... pronto
Roda criada para ibm-watson: filename = ibm_watson-5.3.0-py3-none
-any.whl size = 408872 sha256 = 35f2b1417afffc4170bf8838e701068cb2b
28670436bc07f0453454abbd67635
Armazenado no diretório: /tmp/wsuser/.cache/pip/wheels/21/d9/82/4
ce5b94730bc4f1f7b4c6384f72964b9b8f79fcc125bb8085c
Ibm-watson construído com sucesso
Instalando pacotes coletados: murmurhash, cymem, wasabi, srsly, qtp
y, pashed, plac, catalog, blis, websocket-client, thinc, qtconsol
e, jupyter-console, spacy, jupyter, ibm-watson, assistant-dialog-sk
ill-analysis
Assistant-dialog-skill-analysis-1.2.1 blis-0.4.1 catalog-1.0.0 cyme
m-2.0.5 ibm-watson-5.3.0 jupyter-1.0.0 jupyter-console-6.4.0 murmur
hash-1.0 instalado com sucesso .5 plac-1.1.3 pashed-3.0.5 qtconsol
e-5.1.1 qtpy-1.11.2 spacy-2.3.2 srsly-1.0.5 thinc-7.4.1 wasabi-0.8.
2 websocket-client-1.1.0
```

Em [3]:

```
# Bibliotecas padrão python
import sys , os
import json
import importlib
de coleções import Counter

# Bibliotecas externas python
importam pandas como pd
import numpy as np
import nltk
nltk . download ( 'stopwords' )
nltk . download ( 'punkt' )
import ibm_watson

# Bibliotecas python internos
de assistant_dialog_skill_analysis.utils importar skills_util , lang_utils
de assistant_dialog_skill_analysis.highlighting importação marcador
de assistant_dialog_skill_analysis.data_analysis importação summary_generator
de assistant_dialog_skill_analysis.data_analysis importação divergence_analyz
er
de assistant_dialog_skill_analysis.data_analysis importação similarity_analyz
er
de assistant_dialog_skill_analysis.term_analysis importação chi2_analyzer
de assistant_dialog_skill_analysis.term_analysis importação keyword_analyzer
de assistant_dialog_skill_analysis.term_analysis importação entity_analyzer
de assistant_dialog_skill_analysis.confidence_analysis importação confidence_
analyzer
de assistant_dialog_skill_analysis.inferencing importação inferencer
de assistant_dialog_skill_analysis.experimentation importação data_manipulato
r
```

```
[nltk_data] Baixando palavras de interrupção do pacote para / home
 / wsuser / nltk_data ...
[nltk_data] Descompactando corpora / stopwords.zip.
[nltk_data] Baixando pacote punkt para / home / wsuser / nltk_data
...
[nltk_data] Descompactando tokenizers / punkt.zip.
```

Índice

1. [Parte 1: Análise de dados de treinamento](#)
 - 1.1 [Dados de treinamento de habilidade de diálogo de processo](#)
 - 1.2 [Análise de distribuição de dados](#)
 - 1.3 [Análise de termos - Análise de correlação](#)
 - 1.4 [Análise de termos - Mapa de calor](#)
 - 1.5 [Ambiguidade nos dados de treinamento](#)
2. [Parte 2: Análise do Modelo](#)
 - 2.1 [Análise do Modelo](#)
3. [Parte 3: Análise Avançada](#)
 - 3.1 [Análise usando Limiares de Confiança](#)
 - 3.2 [Destacando a Importância do Termo](#)
 - 3.3 [Análise de Confiança Anormal](#)
 - 3.4 [Análise usando Entidades Correlacionadas por Intenção](#)
4. [Resumo](#)

Parte 1: Análise de dados de treinamento

Configuração: acessar dados de treinamento

Forneça credenciais de acesso para uma habilidade de diálogo existente que você gostaria de analisar.

Tenha seus valores de API Key e Workspace ID à mão

Em [4]:

```
importlib . recarregar ( skills_util )

# Altere a versão da API do assistente, se necessário
# Find Latest -> https://cloud.ibm.com/docs/services/assistant?topic=assistant-r
# release-notes
API_VERSION = '2021-06-14'

# escolha um datacenter para usar
datacenters = {
    'dallas' : ( 'https://api.us-south.assistant.watson.cloud.ibm.com' ,
'https://iam.cloud.ibm.com/identity / token ' ),
    ' washington ' : ( ' https://gateway-wdc.watsonplatform.net/assistant/api '
, ' https://iam.cloud.ibm.com/identity/token ' ),
    ' frankfurt ' : ( 'https://gateway-fra.watsonplatform.net/assistant/api' ,
'https://iam.cloud.ibm.com/identity/token' ),
    'sydney' : ( 'https: // gateway -syd.watsonplatform.net/assistant/api '
, 'https://iam.cloud.ibm.com/identity/token' ),
    'tokyo' : ( 'https://gateway-tok.watsonplatform.net/assistant/api' ,
'https://iam.cloud .ibm.com / identity / token ' ),
    ' london ' : ( ' https://gateway-lon.watsonplatform.net/assistant/api '
, ' https://iam.cloud.ibm.com/identity/token ' ),
    ' staging ' : ( ' https://gateway-s.watsonplatform.net/assistant/api ' ,
' https://iam.cloud.ibm.com/identity/token ' ),
    ' internal ' : ( 'https://api.us-south.assistant.test.watson.cloud.ibm.co
m' , 'https://iam.test.cloud.ibm.com/identity/token' )
}
URL , authenticator_url = datacenters [ 'dallas' ]

# Para ICP (IBM Cloud Private), você pode desativar a verificação SSL alterando
# para True
DISABLE_SSL_VERIFICATION = False

# Por padrão, precisamos apenas da chave da API IAM e do ID do espaço de trabalh
o

# Se você executa o bloco de notas regularmente, pode descomentar as duas linhas
# abaixo
# e comentar a linha depois dele
#iam_apikey = '###'
#workspace_id = '###'
iam_apikey , workspace_id = skills_util . input_credentials ()
conversação = skills_util . retrieve_conversation ( iam_apikey = iam_apikey ,
url = URL ,
api_version = API_VERSION ,
authenticator_url = authenticat
or_url )

#Se você não tem chaves de API baseadas em IAM
#mas tem acesso a um nome de usuário, senha e ID do espaço de trabalho
#Você pode comentar as duas linhas acima e descomentar as linhas abaixo para aut
enticar
# username = 'apikey'
# password = '## #'
# workspace_id = ' ### '
```

```
# talks = skills_util.retrieve_conversation (username = username,  
# password = password,  
# url = URL,  
# api_version = API_VERSION)  
  
conversa . set_disable_ssl_verification ( DISABLE_SSL_VERIFICATION )  
  
espaço de trabalho = skills_util . retrieve_workspace ( workspace_id = workspa  
ce_id ,  
conversação = conversação )
```

Por favor, indique apikey:

Por favor, insira o id do espaço de trabalho:

Escolha o código de idioma correspondente aos dados do seu espaço de trabalho:

Códigos de idioma suportados: en, fr, de, es, cs, it, pt

Em [5]:

```
importlib . reload ( lang_utils )  
LANGUAGE_CODE = "pt" # altera o código do idioma para trabalhar com outros idio  
mas  
lang_util = lang_utils . LanguageUtility ( LANGUAGE_CODE )
```

Em [6]:

```
# Extract usuário espaço de trabalho
workspace_pd , workspace_vocabulary = skills_util . extract_workspace_data (
espaço de trabalho , language_util = lang_util )
entity_dict = conversa . list_entities ( workspace_id ) . get_result ()
entities_list = [ artigo [ 'entidade' ] para artigo em entity_dict [ 'entidades' ] ]

display ( Markdown ( "### Amostra de declarações e intenções" ))
display ( HTML ( workspace_pd . sample ( n = len ( workspace_pd ) se len ( workspace_pd ) < 10 else 10 )
. to_html ( index = False )))
se entity_list :
display ( Markdown ( "### Sample of Entities" ))
display ( HTML ( pd . Trama de dados ( { "Entidade" : entities_list } )
. Amostra ( n = len ( entities_list ) se len ( entities_list ) < 10 else 10 )
. To_html ( index = False )))
```

Amostra de enunciados e intenções

| enunciado | intenção | tokens |
|---|----------------------------|--|
| quais marcas de carro estão hj nas pistas | Carros_10_equipes | [qua, marc, de, carr, esta, hj, nas, pist] |
| me fala do calendario | agenda | [me, fal, do, calendari] |
| informacoes sobre a corrida em são paulo | gp_são_paulo | [informaco, sobr, a, corr, em, são, paul] |
| quais os principais carros | Carros_10_equipes | [qua, os, princip, carr] |
| como adquirir ingressos | gp_são_paulo | [com, adquir, ingress] |
| carros principais | Carros_10_equipes | [princip, carr] |
| pilotos famosos | maiores_ganhadores_titulos | [piloto, famos] |
| qual o inicio da f1 | historia_f1 | [qual, o, inici, da, f1] |
| quais os principais pilotos brasileiros | pilotos_brasileiros | [qua, os, princip, pilot, brasileir] |
| o que é grande premio | grandes_premios | [o, que, e, grand, premi] |

Amostra de Entidades

Entidade

eventos

1.1 Dados de treinamento de habilidades de diálogo de processo

Geramos estatísticas resumidas relacionadas à habilidade e espaço de trabalho fornecidos

Em [7]:

```
importlib . recarregar ( summary_generator )
summary_generator . generate_summary_statistics ( workspace_pd , entity_list )
```

Estatísticas resumidas

| | Característica de dados | Valor |
|---|--|-------|
| 1 | Total de exemplos de usuários | 154 |
| 2 | Intenções Únicas | 15 |
| 3 | Exemplos de usuários médios por intent | 10 |
| 4 | Desvio Padrão da Média | 4 |
| 5 | Número Total de Entidades | 1 |

1.2 Análise de distribuição de dados

Análise de desequilíbrio de classe

Analisamos se o conjunto de dados contém desequilíbrio de classe verificando se o maior intent contém menos do que o dobro do número de exemplos de usuários contidos no menor intent. A presença de desequilíbrio não indica necessariamente um problema, revise a seção de ações abaixo

Em [8]:

```
importlib . reload ( summary_generator )
class_imb_flag = summary_generator . class_imbalance_analysis ( workspace_pd )
```

Desequilíbrio de classe detectado

- Os dados podem ser potencialmente tendenciosos para intents com mais exemplos de usuários
- Por exemplo, Intent <despedida> tem <23> exemplos de utilizadores enquanto que intent <regulamentos> tem apenas <5> exemplos de utilizadores

Distribuição de exemplos de usuários por intenção

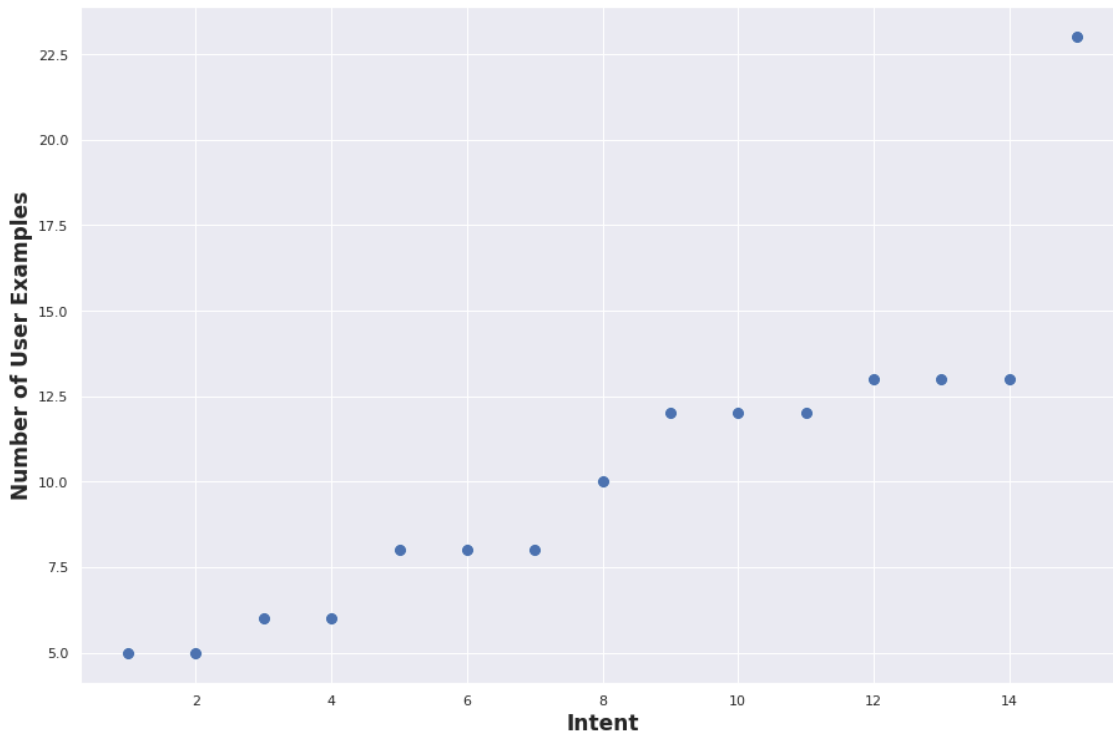
Exibimos a distribuição de intents vs o número de exemplos por intent (classificados pelo número de exemplos por intent) abaixo. Idealmente, não devemos ter grandes variações em termos de número de exemplos de usuários para várias intenções.

Em [9]:

```
importlib . recarregar ( summary_generator )  
summary_generator . scatter_plot_intent_dist ( workspace_pd )
```

Distribuição classificada de exemplos de usuário por intent

findfont: família de fontes ['normal'] não encontrada. Recuando para a DejaVu Sans.



Em [10]:

```
importlib . recarregar ( summary_generator )
summary_generator . show_user_examples_per_intent ( workspace_pd )
```

Distribuição classificada de exemplos de usuário por intenção

| | Intenção | Número de exemplos de usuários |
|----|----------------------------------|--------------------------------|
| 1 | códigos | 5 |
| 2 | pilotos_brasileiros | 5 |
| 3 | sobre_o_nome_formula_1 | 6 |
| 4 | o_que_e_formula_1 | 6 |
| 5 | maiores_ganhadores_titulos | 8 |
| 6 | agenda | 8 |
| 7 | grandes_premios | 8 |
| 8 | historia_f1 | 10 |
| 9 | gp_são_paulo | 12 |
| 10 | Carros_10_equipes | 12 |
| 11 | passos_para_se_tornar_piloto | 12 |
| 12 | menu_assuntos | 13 |
| 13 | acidentes_fatais_vitimas_precoce | 13 |
| 14 | saudação | 13 |
| 15 | despedida | 23 |

Ações para desequilíbrio de classe

O desequilíbrio de classe nem sempre leva a uma precisão menor. Todas as intents (classes), portanto, não precisam ter o mesmo número de exemplos.

1. Para intents como `updateBankAccount` e `addNewAccountHolder` onde a diferença semântica entre eles é mais sutil, o número de exemplos por intent precisa ser um pouco balanceado, caso contrário, o classificador pode favorecer o intent com o maior número de exemplos.
2. Para intents como `greetings` esse são semanticamente distintos de outros intents como `updateBankAccount`, pode ser normal ter menos exemplos por intent e ainda ser fácil para o detector de intent classificar.

Se, durante o teste, parecer que a precisão da classificação da intenção é menor do que o esperado, recomendamos que você examine novamente esta análise de distribuição.

Com relação à distribuição classificada de exemplos por intent, se o número classificado de exemplos do usuário variar muito entre diferentes intents, isso pode ser uma fonte potencial de polarização para detecção de intent. Grandes desequilíbrios em geral devem ser evitados. Isso pode potencialmente levar a uma menor precisão. Se o seu gráfico exibir essa característica, isso pode ser uma fonte de erro.

Para obter mais orientações sobre como adicionar mais exemplos para ajudar a equilibrar sua distribuição, consulte [Intent-Example-Recommendation](https://cloud.ibm.com/docs/services/assistant?topic=assistant-intent-recommendations#intent-recommendations-get-example-recommendations) (<https://cloud.ibm.com/docs/services/assistant?topic=assistant-intent-recommendations#intent-recommendations-get-example-recommendations>).

1.3 Análise de Termos - Análise de Correlação

Recupere os unigramas e bigramas mais correlacionados para cada intenção

Realizamos um teste de significância do qui quadrado usando recursos de contagem para determinar os termos mais correlacionados com cada intenção no conjunto de dados.

A unigram é uma única palavra, enquanto a bigram são duas palavras consecutivas dos dados de treinamento. Por exemplo, se você tiver uma frase como Thank you for your service , cada uma das palavras na frase é considerada unigrama, enquanto termos como Thank you , your service são considerados bigramas.

Se você vir termos como hi , hello correlacionados com uma greeting intenção que seria razoável. Mas se você vir termos como table , chair correlacionados com a greeting intenção, isso seria anômalo. Uma varredura dos unigramas e bigramas mais correlacionados para cada intent pode ajudá-lo a detectar possíveis anomalias em seus dados de treinamento.

Nota : Ignoramos as seguintes palavras comuns por consideração an, a, in, on, be, or, of, a, and, can, is, to, the, i

Em [11]:

```
importlib . recarregar ( chi2_analyzer )  
unigram_intent_dict , bigram_intent_dict = chi2_analyzer . get_chi2_analysis  
( workspace_pd , lang_util = lang_util )
```

Análise Qui-quadrado

| | Intenção | Unigramas Correlacionados | Bigramas Correlacionados |
|----|-----------------------------------|--|---|
| 1 | despedida | grat, obrig, val, legal, comeu | obrigado, legal fui, ate amanh, comi fora, val mesm |
| 2 | saudação | bom, belz, ola, tard, boa | bom dia, boa tard, boa noit |
| 3 | acidentes_fatais_vitimas_precoces | trágico, gordo, falec, acidente, morte | brasileir mort, acidente trágico, trágico f1, qua acidente, gordo f1 |
| 4 | menu_assuntos | poss, encontr, pergunt, voc, aqu | esse bot, tópico encontr, tenh aqu, voc fal, encontr aqu |
| 5 | passos_para_se_tornar_piloto | precis, rasgado, par, piloto, ser | piloto f1, piloto rasgado, fórmula piloto, quer ser, ser piloto |
| 6 | Carros_10_equipes | equip, marc, das, hj, carr | qua marc, princip carr, carr das, carr corr, sobr carr |
| 7 | gp_são_paulo | paul, sp, interlag, ingresso, mapa | gp samp, ingress pra, etap interlag, são paulo, gp sp |
| 8 | historia_f1 | qual, etap, foi, primeir, histor | comec formul, comec f1, histor formul, foi primeir, primeir etap |
| 9 | grandes_premios | signif, soa, gp, grand, premi | qua soa, signif gp, soa grand, sobr grand, grand premi |
| 10 | agenda | sei, dat, calendari, evento, agend | onde sei, qual agend, sab agend, sab calendari, sobr evento |
| 11 | maiores_ganhadores_titulos | dentr, campeo, ganh, famos, titul | dentr formul, campeo formul, quem ganh, titul formul, ganh titul |
| 12 | o_que_e_formula_1 | f1, quer, sab, sobr | quer sab, sab sobr, sobr formul, sobr f1 |
| 13 | sobre_o_nome_formula_1 | surg, saiu, vem, deu, nom | deu nom, cham formul, quem deu, vem ser, nom formul |
| 14 | pilotos_brasileiros | quem, piloto, sao, conhec, brasileir | sobr piloto, brasileir conhec, quem sao, sao piloto, piloto brasileir |
| 15 | códigos | qua, as, regr, requisit, regul | requisit formula, sao regul, sobr regul, tem regul, regul f1 |

Ações para correlações anômalas

Se você identificar termos correlacionados incomuns / anômalos como: números, nomes etc., que não devem ser correlacionados com uma intenção, leia o seguinte:

- **Caso 1** : se você vir nomes aparecendo entre unigramas ou bigramas correlacionados, adicione mais variações de nomes para que nenhum nome específico seja correlacionado
- **Caso 2** : se você vir números específicos como 1234 entre unigramas ou bigramas correlacionados e não forem úteis para o caso de uso, remova ou mascare esses números dos exemplos
- **Caso 3** : se você vir termos que nunca devem ser correlacionados a esse intent específico, considere adicionar ou remover termos / exemplos para que os termos específicos do domínio sejam correlacionados com o intent correto

1.4 Análise de Termos - Mapa de Calor

Um mapa de calor de termos é um método pelo qual podemos visualizar quais termos ou palavras ocorrem com frequência em cada intenção. As linhas são os termos e as colunas são as intenções.

Por padrão, mostramos apenas as 30 principais intents com o maior número de exemplos de usuários na análise. Este número pode ser alterado, se necessário.

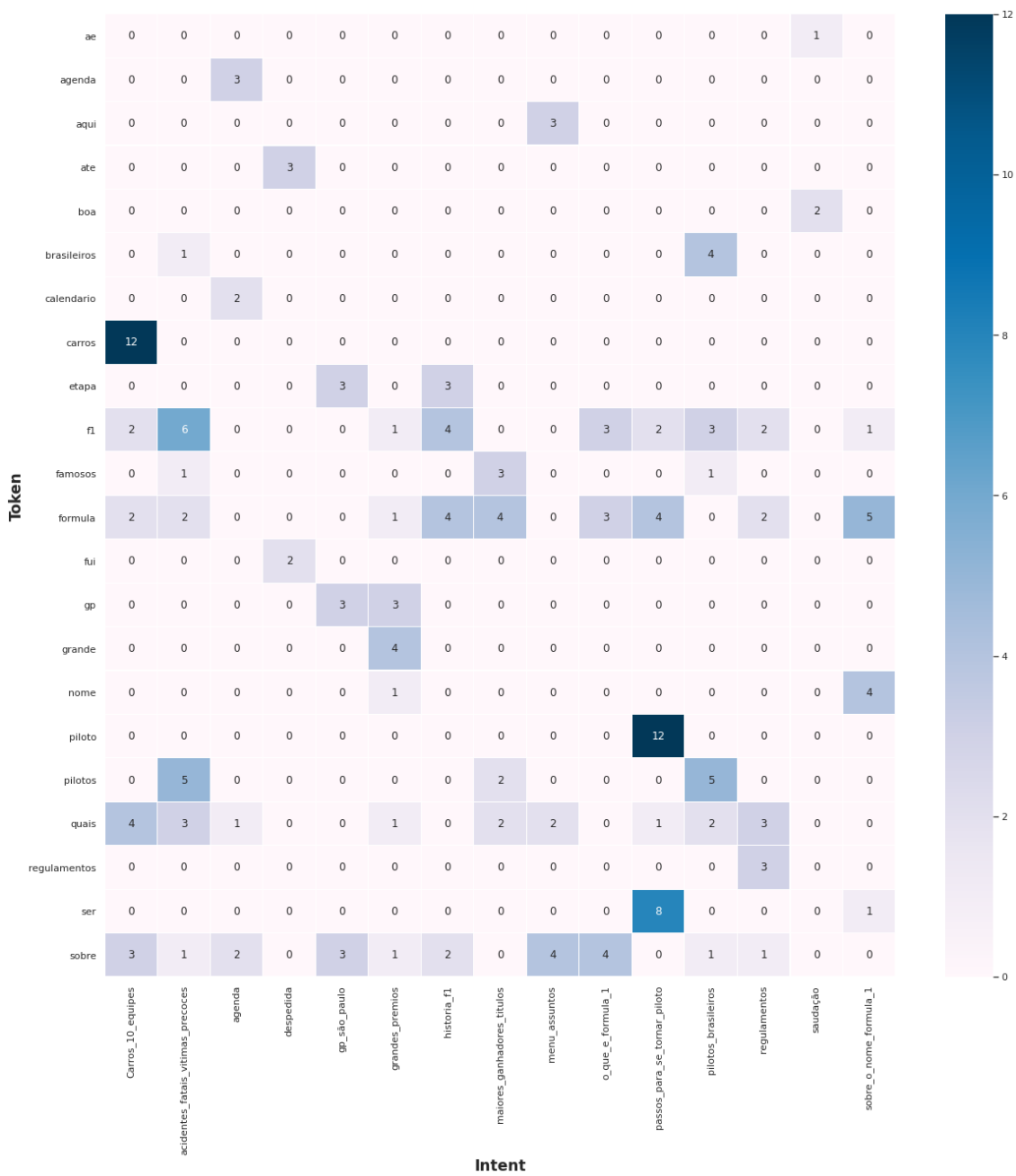
Em [12]:

```
importlib . recarregar ( keyword_analyzer )

INTENTS_TO_DISPLAY = 30 # Número total de
intents para exibição MAX_TERMS_DISPLAY = 30 # Número total de termos para e
xibir

intent_list = []
analisador de palavras-chave . seaborn_heatmap ( workspace_pd , lang_util , IN
TENTS_TO_DISPLAY , MAX_TERMS_DISPLAY , intent_list )
```

Freqüência de token por intent



Análise de termos para lista de intenções personalizadas

Se você deseja ver a análise de termos para intents específicos, sintá-se à vontade para adicionar esses intents à lista de intents. Isso deve gerar um mapa de calor de termo personalizado. Por padrão, mostramos os 30 principais tokens, mas isso pode ser alterado se necessário

Em [13]:

```
importlib . reload ( keyword_analyzer )
# intent_list = ['intent1', 'intent2', 'intent3']
intent_list = []

MAX_TERMS_DISPLAY = 20 # Número total de termos a serem exibidos

if intent_list :
    keyword_analyzer . seaborn_heatmap ( workspace_pd , INTENTS_TO_DISPLAY , M
AX_TERMS_DISPLAY , intent_list )
```

Ações para termos anômalos no mapa de calor

Se você notar quaisquer termos ou palavras que não devem estar frequentemente presentes em uma intent, considere modificar os exemplos dessa intent

1.5 Ambiguidade nos dados de treinamento

Descubra termos possivelmente ambíguos com base na correlação de recursos

Com base na análise do qui-quadrado acima, geramos pares de intenções cujos unigramas e bigramas correlacionados se sobrepõem. Isso nos permite vislumbrar quais unigramas ou bigramas podem causar confusão potencial na detecção de intenção.

A. Pares de Intenção Principais cujos unigramas correlacionados se sobrepõem

Em [14]:

```
importlib . recarregar ( chi2_analyzer )
ambiguous_unigram_df = chi2_analyzer . get_confusing_key_terms ( unigram_inten
t_dict )
```

| Pares de Intenção | Termos |
|-------------------|--------|
|-------------------|--------|

| | |
|---|--------|
| <passos_para_se_tornar_piloto, pilotos_brasileiros> | piloto |
|---|--------|

B. Pares de Intenção Principais cujos bigramas correlacionados se sobrepõem

Em [15]:

```
importlib . recarregar ( chi2_analyzer )
ambiguous_bigram_df = chi2_analyzer . get_confusing_key_terms ( bigram_intent_
dict )
```

Não há ambigüidade com base nos 5 principais termos da análise chi2

C. Verificador de sobreposição para intenções específicas

Em [18]:

```
# Adicione intent específico ou pares de intent para os quais você gostaria de v
er sobreposição
importlib . reload ( chi2_analyzer )
intent1 = 'agenda'
intent2 = 'grandes_premios'
chi2_analyzer . chi2_overlap_check ( ambiguous_unigram_df , ambiguous_bigram_df
, intent1 , intent2 )
```

Pares de Intenção Termos

Descubra declarações ambíguas em intents

A análise a seguir mostra exemplos de usuários que são semelhantes, mas se enquadram em diferentes intents.

Em [19]:

```
importlib . recarregar ( similarity_analyzer )
similar_utterance_diff_intent_pd = similarity_analyzer . ambiguous_examples_an
alysis ( workspace_pd , lang_util )
```

| Intenção 1 | Enunciado 1 | Intenção 2 | Enunciado 2 | pontuação de similaridade |
|---------------|------------------------------|-------------------|-------------------------|------------------------------|
| saudação | fala | despedida | falou | 1.000000 |
| agenda | quero saber sobre eventos | o_que_e_formula_1 | quero saber sobre f1 | 0,714286 |

Ações para ambigüidade nos dados de treinamento

Pares de intenção ambíguos

Se você vir termos que estão correlacionados com mais de 1 intenção, revise se isso parece anômalo com base no caso de uso para essa intenção. Se parecer razoável, pode não ser um problema.

Enunciados ambíguos em intents

1. **Enunciados duplicados** : para enunciados duplicados ou quase idênticos, remova aqueles que parecem desnecessários
2. **Enunciados semelhantes** : para enunciados semelhantes, reveja o caso de uso para essas intents e certifique-se de que não são acréscimos acidentais causados por erro humano na criação dos dados de treinamento

Referência para mais informações sobre a entidade: [Documentação da Entidade](https://cloud.ibm.com/docs/services/assistant/services/assistant?topic=assistant-entities)
(<https://cloud.ibm.com/docs/services/assistant/services/assistant?topic=assistant-entities>)

Para uma análise mais aprofundada relacionada a possíveis conflitos em seus dados de treinamento entre intents, experimente o recurso de detecção de conflito no Watson Assistant: [Documentação de resolução de conflito](https://cloud.ibm.com/docs/services/assistant?topic=assistant-intents#intents-resolve-conflicts) (<https://cloud.ibm.com/docs/services/assistant?topic=assistant-intents#intents-resolve-conflicts>)

Parte 2: Análise do modelo

Analise sua habilidade de diálogo do Watson Assistant existente com a ajuda de um conjunto de teste.

Configuração: Carregar dados de teste

Faça upload de um conjunto de teste no formato csv. Cada linha do arquivo deve ter apenas User_Input<comma>Intent

Um exemplo seria

```
hello how are you<comma>Greeting  
I would like to talk to a human<comma>AgentHandoff
```