



pgembed

Enquanto extensões como o pgvector preparam o PostgreSQL para armazenar e consultar dados vetoriais, o processo de gerar esses vetores — a vetorização (*embedding*) — ainda precisa de ser executado separadamente.

Normalmente, este processo exige que uma aplicação ou *pipeline* comunique com um modelo de *Machine Learning* externo, como os oferecidos pela OpenAI, ou localmente, com modelos disponíveis no Ollama. O que pode exigir um conhecimento técnico mais amplo da equipa de dados ou o envolvimento de diversos profissionais com competências variadas.

Em muitos dos casos, projetos focados na validação do *Minimum Viable Product (MVP)* optam por desenvolver funções no próprio PostgreSQL que consomem serviços de vetorização e facilitam essa atividade.

Contudo, é comum que as boas práticas, como o uso de dados em lotes para a redução do tráfego de rede, gestão do consumo de memória, configurações de segurança e mecanismos de resiliência, sejam negligenciadas nesta abordagem.

Com o objetivo de fornecer um conjunto de funções para vetorização de dados no PostgreSQL, com foco em desempenho, segurança e resiliência, a Tecnisys desenvolveu a extensão pgembed.

Funções de Vetorização

Após instalar e criar a extensão pgembed, serão disponibilizadas no esquema `pgembed` da base de dados diversas funções para a vetorização de dados, organizadas pelo fornecedor de API de *embedding* e pelo tipo de dados de entrada:

Fornecedor	Linha de Texto Única	Conjunto de Linhas	Tabela
Ollama (local)	<code>embed_ollama</code>	<code>embed_batch_ollama</code>	<code>embed_table_ollama</code>
OpenAI	<code>embed_openai</code>	<code>embed_batch_openai</code>	<code>embed_table_openai</code>
Custom APIs	<code>embed_custom</code>	<code>embed_batch_custom</code>	<code>embed_table_custom</code>

Tipos de Funções:

- **Linha de Texto Única** (`embed_*`): Gera o *embedding* para uma única linha de texto.

Por exemplo:

```
SELECT pgembed.embed_openai('Hello, world!', 'text-embedding-3-small');
```

- **Conjunto de Linhas** (`embed_batch_*`): Processa múltiplas linhas de texto e devolve uma lista de vetores.

Por exemplo:

```
SELECT * FROM pgembed.embed_batch_openai(ARRAY['Hello, world!', 'Hello, PostgreSQL!'], 'text-embedding-3-small');
```

- **Tabela** (`embed_table_*`): Lê a coluna de texto de uma tabela e atualiza automaticamente a coluna de *embeddings* correspondente, eliminando a necessidade de operações manuais de atualização.

Por exemplo:

```
SELECT * FROM pgembed.embed_table_openai(  
    'public',          -- schema  
    'tb_documents',    -- table name  
    'content',         -- content column  
    'embedding',       -- embedding column  
    FALSE,             -- regenerate: only update NULL embeddings (default:  
FALSE)  
    1000,              -- batch_size  
    'text-embedding-3-small' -- model  
);
```

Parâmetros Avançados

Além dos parâmetros obrigatórios, as funções da extensão pgembed possuem diversos parâmetros avançados extremamente úteis, tais como:

- Timeout da requisição (`timeout` , default 60s)
- Se o certificado SSL deve ser verificado (`verify_ssl` , default TRUE)
- Se o texto deve ser truncado caso exceda o tamanho do contexto (`truncante` , default TRUE, para modelos do Ollama)

- JSON para opções avançadas de cada modelo (`options` , para modelos do Ollama)
- Número de dimensões dos *embeddings* gerados (`dimensions` , para modelos da OpenAI)
- Formato de codificação dos *embeddings* devolvidos (`encode_format` , para modelos da OpenAI)

Segurança e Resiliência

No que diz respeito à segurança e resiliência, a extensão pgembed disponibiliza parâmetros que podem ser configurados ao nível do utilizador, sessão e base de dados, tais como:

- `pgembed.openai_api_key` ou `pgembed.custom_api_key` para definir a API key utilizada na requisição do serviço
- `pgembed.url_allowlist` para definir a lista de URLs autorizados. Por padrão: localhost, 127.0.0.1, 0.0.0.0 (qualquer porta), *.openai.com e api.openai.com
- `pgembed.max_retries` , `pgembed.initial_backoff_ms` , `pgembed.max_backoff_ms` para controlo de tentativas
- `pgembed.circuit_breaker_threshold` e `pgembed.circuit_breaker_reset_timeout_s` para evitar que falhas em cascata bloqueiem temporariamente solicitações para serviços indisponíveis

Fonte: <https://github.com/tecnisys/pgembed>

A instalar o pgembed via PgSmart CLI

Comando

```
bash title="Terminal input" pgsmart install
```

1. Selecione a Opção Base de Dados.
2. Selecione a Opção `Extensões` .
3. Selecione a Extensão `Integração com Text Embedding APIs (pgembed)` .
4. Indique a versão Majoritária do PostgreSQL.
 - 4.1 Indique a Release da versão Majoritária do PostgreSQL.
5. Confirme a instalação da extensão:

5.1. Confirme se deseja instalar/atualizar o repositório dos pacotes do PostgreSYS (esta instalação é necessária para dar continuidade, caso o repositório ainda não esteja configurado).

5.2. Caso tenha optado por instalar/atualizar o Repositório de Pacotes, indique a URL do Repositório de Pacotes.

** NOTE**

A opção de instalação do Repositório de Pacotes não é realizada se os repositórios do pgsys já estiverem presentes ou mapeados na máquina.

```
[pgsmart@pgsmart-CentOS8-1 ~]$ pgsmart install

AGENTES REGISTRADOS

? Escolha um PgSmart Agent:

  ALIAS      HOST      PORTA
(Use arrow keys or type to search)
> 192.168.56.236 localhost 4432
  192.168.56.237 192.168.56.237 4432
  192.168.56.238 192.168.56.238 4432

Registrar
Sair

SELEÇÃO DE SERVIÇOS

i Agente selecionado: 192.168.56.236 (localhost:4432)

? Serviços para instalação: (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
o Administração e Operação do Ambiente PostgreSYS
>● Banco de Dados
o Gerenciamento de Backups
o Pool de Conexões
o Alta Disponibilidade
o Observabilidade

SELEÇÃO DE TIPOS DE COMPONENTES

i Agente selecionado: 192.168.56.236 (localhost:4432)

? Tipos de componentes do serviço de Banco de Dados: (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
o Sistema de Gerenciamento de Bancos de Dados
>● Extensões de Banco de Dados

SELEÇÃO DE COMPONENTES

i Agente selecionado: localhost (localhost:4432)

? Componentes do serviço de Banco de Dados: (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
o Georreferenciamento (PostGIS)
o Busca por Similaridade Vetorial (pgvector)
>● Integração com Text Embedding APIs (pgembed)

SELEÇÃO DA VERSÃO BASE DO POSTGRESYS

i Agente selecionado: 192.168.56.236 (localhost:4432)

? Versão majoritária do PostgreSQL: 16
? Release da versão majoritária 16: 16.8

Δ Os binários do PostgreSQL 16.8 serão instalados
com o(s) componente(s) escolhido(s)!

? Deseja continuar com a instalação? Yes

REPOSITÓRIO DE PACOTES POSTGRESYS

i Agente selecionado: 192.168.56.236 (localhost:4432)

? Deseja atualizar o repositório de pacotes da Plataforma PostgreSYS? No

REPOSITÓRIO DE PACOTES POSTGRESYS

i Agente selecionado: 192.168.56.236 (localhost:4432)

? Deseja atualizar o repositório de pacotes da Plataforma PostgreSYS? No

INSTALAÇÃO DE SERVIÇOS E COMPONENTES

i Agente selecionado: 192.168.56.236 (localhost:4432)

✓ Preparação para a instalação do componente realizada com sucesso!

✓ Instalação realizada com sucesso (Extensões de Banco de Dados [Integração com Text Embedding APIs (pgembed)]):

CONFIGURAÇÕES DO HOST

i Agente selecionado: 192.168.56.236 (localhost:4432)

i Executando as configurações finais
✓ Registro do diretório de binários realizado com sucesso!

✓ Configurações realizadas com sucesso!

AJUDA RÁPIDA

i Agente selecionado: 192.168.56.236 (localhost:4432)

i Conecte-se ao banco de dados e execute o comando abaixo para habilitar a extensão de Integração com Text Embedding APIs (pgembed):
CREATE EXTENSION pgembed;

Até logo!
```

Figura 6 - Instalação da extensão pgembed interativa

6. Ligue-se à base de dados e execute o comando abaixo:

Terminal input

```
CREATE EXTENSION pgembed;
```

A instalar o pgembed via PgSmart Web

1. Selecione o separador [Início/Gerenciar Ambientes/Serviços](#).

Serão apresentados todos os serviços instalados em todos os ambientes.

2. Clique em [Instalar](#).

3. Selecione o [Ambiente](#) onde o Serviço será instalado.

4. Selecione o serviço [Base de Dados](#).

5. Selecione a [Versão Base](#) do PostgreSQL [Ex: 16].

6. Selecione a [Versão do PostgreSQL](#) [Ex: 16-2].

7. Clique em [Avançar](#).

8. Selecione a Extensão [Integração com Text Embedding APIs \(pgembed\)](#).

9. Indique se deseja inicializar a instância de base de dados.

10. Selecione o [endereço IP](#) de um ou mais Agentes do PgSmart.

11. Clique em [Instalar](#).

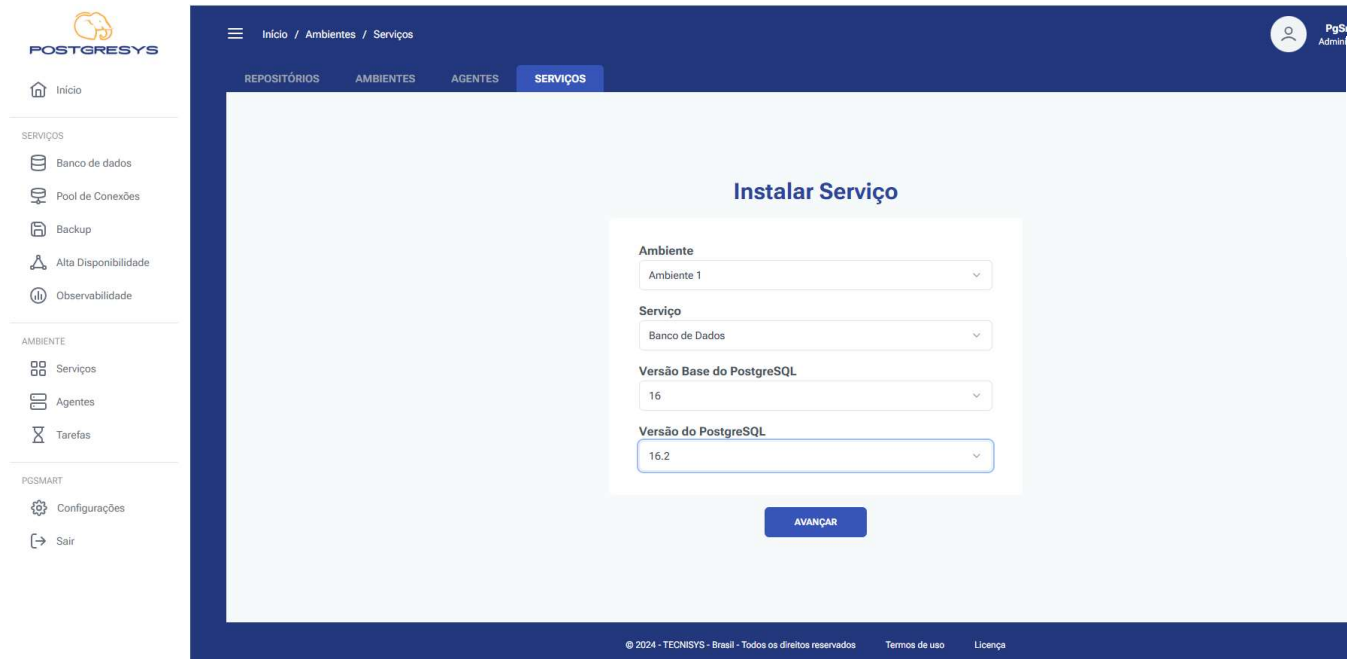


Figura 1 - Instalação do Serviço de Base de Dados

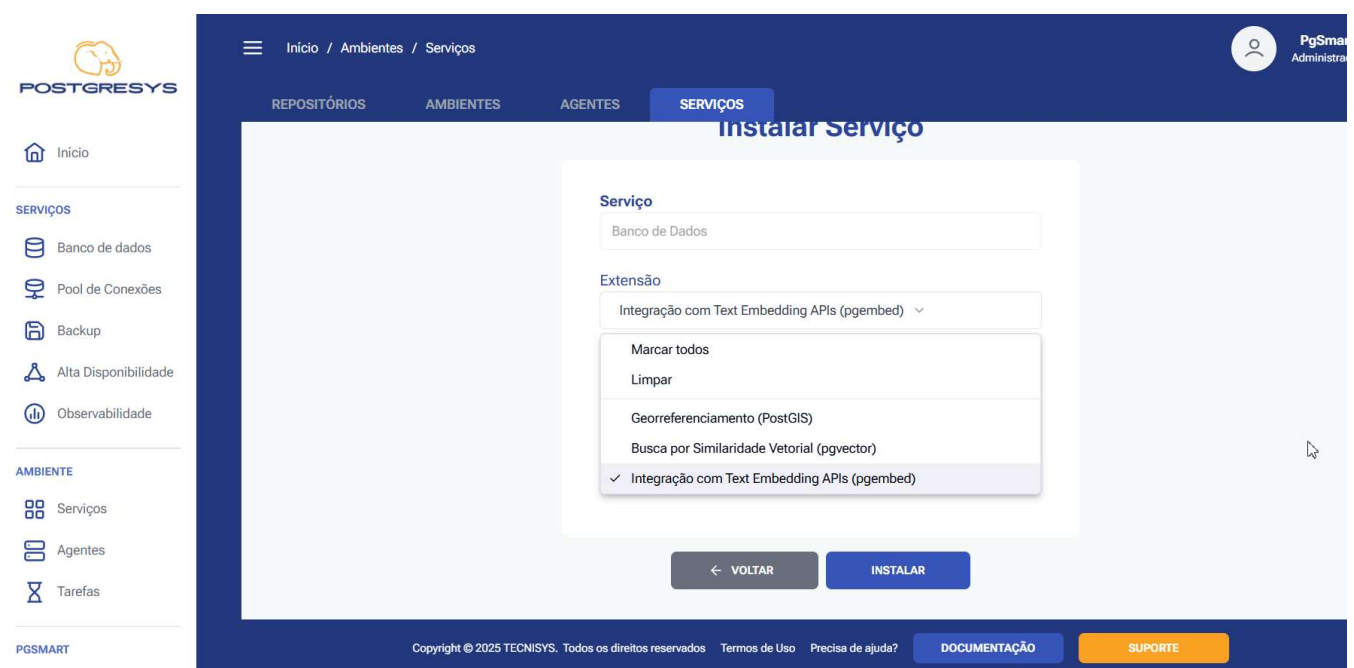


Figura 2 - Instalação do Serviço de Base de Dados

Início / Ambientes / Serviços

PgS Admin

REPOSITÓRIOS

AMBIENTES

AGENTES

SERVIÇOS

Instalar Serviço

192.168.56.236

✓

Preparação para a instalação dos componentes realizada com sucesso!

✓

Instalação realizada com sucesso (Sistema de Gerenciamento de Bancos de Dados)!

✓

Instalação realizada com sucesso (Extensões de Banco de Dados)!

i

Executando as configurações finais do ambiente

✓

Registro do diretório de binários realizado com sucesso!

✓

Configurações do ambiente realizadas com sucesso!

© 2024 - TECNISYS - Brasil - Todos os direitos reservados

Termos de uso

Licença

Figura 3 - Instalação do Serviço de Base de Dados