

[Job Search documentation](https://cloud.google.com/talent-solution/job-search/) (<https://cloud.google.com/talent-solution/job-search/>)

[Documentation](#)

Job Search parameters: Best practices

Job Search configurations

Factors that affect search results

Together, "Featured Job Search", "Enable Broadening", and "Disable Keyword Match" have a significant impact on the number and relevance of jobs returned to the job seeker. The most appropriate configuration of these three factors depends on your business needs. The best method for determining your optimal configuration is to apply different test scenarios and evaluate the outcomes during a testing phase.

1. **Featured job search:** Use featured job searches to assign promotional values to individual jobs. This allows you to highlight jobs that are important to your business needs. See the [featured job search](https://cloud.google.com/talent-solution/job-search/docs/featured-jobs) (<https://cloud.google.com/talent-solution/job-search/docs/featured-jobs>) documentation for best practices and implementation details.
2. **`disableKeywordMatch`**
(<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/projects.jobs/search>)
: This parameter allows Job Search to return keyword-based matches to a job seeker's query in addition to relevant matches. The default setting is `false`. Setting this parameter to `true` disables the keyword match, so fewer jobs (only those determined to be relevant by the ML feature) are returned.
3. **`enableBroadening`**
(<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/projects.jobs/search>)
: Use this parameter to expand the job seeker's query by relaxing their stated restrictions

on location and job categories. The default setting is `false`. Enabling this parameter increases the number of search results returned but may decrease the relevance of the overall results set to the job seeker.

Search configuration outcomes

To return only the most relevant jobs: Set `disableKeywordMatch` to `true` and `enableBroadening` to `false`. This improves the API's relevance-related performance metrics since only the relevant jobs are returned. However, fewer jobs overall are returned in the search results.

To return a higher number of jobs including both relevant and keyword matched results: Set `disableKeywordMatch` to `false` and `enableBroadening` to `false`. The keyword-matched results are listed after the relevant jobs in the results. Query expansion results are not returned.

To return a higher number of jobs including both relevant and query expanded jobs: Set `disableKeywordMatch` to `true` and `enableBroadening` to `true`. The job seeker's query is expanded to include related job categories and nearby locations. These additional jobs are listed after the relevant results. Keyword-based matches are not returned.

To return the highest possible number of jobs: Set `disableKeywordMatch` to `false` and `enableBroadening` to `true`. Job Search returns the most relevant jobs at the top of the search results, followed by keyword matched jobs and query expanded jobs (by location, job category, and so on). This maximizes the total number of jobs returned.

disableKeywordMatch enableBroadening Outcome

-	-	Returns a higher number of jobs including both relevant and keyword matched results (but assume featured job search is set to <code>false</code>).
+	+	Returns a higher number of jobs including both relevant and query expanded jobs (but assume featured job search is set to <code>false</code>).
-	+	Returns the highest number of jobs (but assume featured job search is set to <code>false</code>).
+	-	Returns only the most relevant jobs (but assume featured job search is set to <code>false</code>).

`Request.page_size`

For best performance and to avoid latency, set up the results page to display 20 or fewer jobs at a time. See the [search overview](#)

(https://cloud.google.com/talent-solution/job-search/docs/search#pagination_optional) page for more information on implementing pagination.

`customAttributes`

`customAttributes` gives you the flexibility to assign additional values to your jobs according to your business needs (for example, GPA scores) and use these values to filter the results.

Note: Avoid setting up any `customAttributes` values that are related to location. Job Search's existing location filter has a built-in algorithm that automatically detects location information related to the job seeker's search query.

Location field

A detailed overview of the `location` field can be found on the [location fields](#)

(https://cloud.google.com/talent-solution/job-search/docs/jobs#location_fields) page. Providing the job's street address instead of GPS coordinates in the `address` field helps Job Search improve location detection and search relevance.

`locationFilter`

Use the `regionCode` field if a single company has multiple job openings in different geographic regions. Assigning a `regionCode` to each listing ensures that a search query returns jobs only in the job seeker's desired location rather than global results matching the search query. For example, a search on the location keyword "Cambridge" without a `regionCode` in place returns results from both Cambridge, UK and Cambridge, MA, USA. This decreases search relevance.

`region_code` and `language_code`

These two fields allow Job Search to use localised search logic in different geographic regions (for example, "lorry driver" in `en_GB` as opposed to "truck driver" in the United States). Set `request.filters.location_filters.region_code` to match the geographic location that the

user is searching (such as the United Kingdom) and `request.filters.language_code` to the appropriate language code for that region (`en_GB` in this case).

`radiusInMiles`

This parameter sets a radius (in miles) around a job seeker's indicated location. Job Search returns results inside this geographic range. How this distance is applied to the geography of the search results depends on the type of location information entered. If the job seeker enters a street address, the `radiusInMiles` distance is set from a single point. If the job seeker enters a city, Job Search applies a bounding box around the city limits and calculates distance from the box edges. If the user enters only a state or country, `radiusInMiles` is ignored.

Make sure that the mileage radius is as small as possible. Setting the mileage to a larger range causes Job Search to return results that may be outside the job seeker's desired location, decreasing relevance. For example, searching for jobs in New York City with `radiusInMiles` set to 100 miles returns results in both New Jersey and Upstate New York. Keeping the radius as small as possible increases the relevance of the results.

`postingExpireTime`

This parameter sets the length of time that the job posting remains active before it's removed from search results. By default, CTS removes jobs 30 days after the creation time (UTC time).

`Job_employment_type`

This is not a required field, however using `job_employment_type` increases the relevance of the search results.

Job Search configurations: Custom ranking

Featured job search allows you to influence a user's search results by highlighting jobs based on a single variable (`promotionValue`). See the [featured job search](https://cloud.google.com/talent-solution/job-search/docs/featured-jobs) (<https://cloud.google.com/talent-solution/job-search/docs/featured-jobs>) documentation for details.

Custom ranking allows you to influence search results based on multiple variables, offering more granular control over the rankings. This feature is useful in applications that require balancing relevance with economic interests, such as a multi-tiered Cost-per-Click (CPC)

subscriber system. Influence over how jobs are ranked on top of the original relevance score is based on two variables: `rankingExpression` and `importanceLevel`.

- **rankingExpression:** This variable controls how jobs are ranked based on their calculated relevance scores. 'rankingExpression' must be set to 'filterable' in order for Job Search to index the parameter.
- **importanceLevel:** This parameter sets the importance level of a job's ranking position when it's returned in search results. There are six possible levels: Unspecified, **NONE**, **LOW**, **MID**, **HIGH** and **EXTREME**. Setting the value to **EXTREME** causes all other API-generated relevance factors to be ignored, so use this value sparingly. Jobs set to **EXTREME** are returned at the top of the job seeker's query instead of the most relevant jobs.
- **Featured job search versus custom ranking:** A **featured job search** is most useful for promoting a single category of jobs above the relevance ranking, for example jobs at a certain company. If you need to rank jobs according to multi-level CPC (Cost Per Click) variables in addition to the relevance ranking, **custom ranking** is a better choice.

Commute search

Commute search (<https://cloud.google.com/talent-solution/job-search/docs/search-commute>) allows job seekers to search for jobs based on commute time. To enable it, include a `CommuteFilter` object in the `JobQuery.commuteFilter` field. The `CommuteFilter` calculates commute time using a job seeker's indicated commute method, travel duration, and start coordinates. Job seekers must also select either `roadTraffic` (`TRAFFIC_FREE` or `BUSY_HOUR`) or `departureTime` to include in the time calculation. See the Commute Search implementation (<https://cloud.google.com/talent-solution/job-search/docs/commute-search>) and how-to (<https://cloud.google.com/talent-solution/job-search/docs/search-commute>) pages for details.

Commute search results are based on historical and aggregated data rather than live traffic conditions. The `departureTime` traffic conditions are calculated from average traffic conditions at the specified time of day. The `BUSY_HOUR`/`TRAFFIC_FREE` options under `roadTraffic` are average traffic conditions at morning rush hour and midnight, respectively. Users receive the same commute search results no matter what time of day they send a query.

Multi-tenancy (optional)

Job Search supports **tenants** as a middle organizational layer between a Google Cloud Platform project and any data uploaded to it. Tenants prevent data from being shared across tenancy barriers, allowing you to isolate subsets of your data without the need for multiple projects. Multi-tenancy is useful in situations where you have multiple customers and don't want to share data between them, but would like to maintain a single GCP project for internal billing and reporting. For example:

- Job site providers building job sites for organizations with multiple subsidiary companies.
- Hiring agencies building applicant tracking systems for multiple businesses.

Each project is assigned a single default tenant ID. You can implement multi-tenancy by [creating more than one tenant](https://cloud.google.com/talent-solution/job-search/docs/tenants) (https://cloud.google.com/talent-solution/job-search/docs/tenants) within a given project.

Tenants are fully isolated from one another. All APIs ask for only a single tenant to prevent data being queried across multiple tenants in the same API call. The machine learning algorithms similarly treat tenants as discrete units and do not cross tenancy barriers. A project can support as many tenants as required.

Security

CTS provides very light tenant support. You are responsible for creating tenants, assigning tenant IDs, and providing the correct tenant ID when making a request. CTS verifies that the tenant ID is owned by a given project and retrieves data from the tenant provided. Any additional security to catch unauthorized access must be managed in your backend system.

Data management & error handling

Data integrity

1. **Uploading jobs:** Data issues can prevent jobs from being uploaded to Job Search. Please see the [HTTP Response Codes](https://cloud.google.com/talent-solution/job-search/docs/http-response-codes) (https://cloud.google.com/talent-solution/job-search/docs/http-response-codes) page for a list of error codes. Common examples include:

- Job locations are incorrect, so the request can't be resolved.
- Company or Job fields do not exist, so a bad request is returned.

There are three main options for troubleshooting job uploading issues:

- Check the logging from your back end.
- Check the CTS [management tool](https://cloud.google.com/talent-solution/docs/management-tools) (<https://cloud.google.com/talent-solution/docs/management-tools>) for data logging.
- Set up the [Stackdriver Monitoring Tool](https://cloud.google.com/monitoring/docs/) (<https://cloud.google.com/monitoring/docs/>) on Cloud Console to collect data on metrics, events, and metadata.

2. **Indexing jobs:** Job Search is designed to index all of your uploaded jobs within a set period of time. However, you may have quota restrictions on your end. Be sure to check your system for restrictions on indexing requests before sending jobs to CTS.

Self-inflicted DDoS attacks

Error handling

API services provided over the internet can have intermittent connection failure, a prolonged outage, sudden service maintenance and other events that require a client application to retry the API request. Make sure to design the retry with network friendly behavior, for example [exponential backoff](https://cloud.google.com/storage/docs/exponential-backoff) (<https://cloud.google.com/storage/docs/exponential-backoff>).

Quota limits

Avoid sending traffic higher than your provisioned quota, especially **far** higher than your provisioned quota. Otherwise, your traffic may be classified as malicious and therefore blocked.

Deduping

Duplicate jobs negatively impact a job seeker's search experience. Job Search includes two features to minimize duplicates:

1. **Create jobs:** If you try to create 2+ jobs with the following criteria, the record is rejected and a 4xx error is returned:
 - same `companyName`, AND
 - same `job_req_id`, AND
 - same location/`languageCode`

2. **Search for jobs:** Job Search surfaces jobs that are relevant to job seeker's search query. A built-in feature of the relevance algorithm makes sure that any returned jobs are diversified, preventing nearly identical jobs from showing up next to each other in search results.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (https://creativecommons.org/licenses/by/4.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (https://www.apache.org/licenses/LICENSE-2.0). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (https://developers.google.com/terms/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated December 13, 2019.