



## Natural Language Processing

# SDK Reference

Issue 01  
Date 2020-01-11

**Copyright © Huawei Technologies Co., Ltd. 2020. All rights reserved.**

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

## **Trademarks and Permissions**



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

## **Notice**

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

## **Huawei Technologies Co., Ltd.**

Address: Huawei Industrial Base  
Bantian, Longgang  
Shenzhen 518129  
People's Republic of China

Website: <https://www.huawei.com>

Email: [support@huawei.com](mailto:support@huawei.com)

---

# Contents

---

<b>1 SDK Introduction</b>	<b>1</b>
<b>2 Authentication</b>	<b>3</b>
<b>3 Obtaining SDKs</b>	<b>4</b>
<b>4 Obtaining a Project ID</b>	<b>5</b>
<b>5 Using the NLP Fundamentals SDK</b>	<b>6</b>
5.1 Initializing the Client	6
5.2 Calling NLP Fundamentals APIs	6
5.2.1 Word Segmentation	7
5.2.2 NER (Basic Edition)	8
5.2.3 NER (Domain-specific Edition)	9
5.2.4 Text Similarity	11
5.2.5 Sentence Vector	12
5.3 Accessing the Client via HTTP Proxies	13
<b>6 Using the LU SDK</b>	<b>14</b>
6.1 Initializing the Client	14
6.2 Calling LU APIs	14
6.2.1 Sentiment Analysis (Basic Edition)	14
6.2.2 Sentiment Analysis (Domain-specific Edition)	15
6.2.3 Text Classification	17
6.2.4 Intent Understanding	18
6.3 Accessing the Client via HTTP Proxies	20
<b>7 Using the LG SDK</b>	<b>21</b>
7.1 Initializing the Client	21
7.2 Calling LG APIs	21
7.2.1 Text Summarization	21
7.3 Accessing the Client via HTTP Proxies	23
<b>8 Using the MT SDK</b>	<b>25</b>
8.1 Initializing the Client	25
8.2 Calling MT APIs	25
8.2.1 Text Translation	25

---

8.2.2 Language Detection.....	27
8.3 Accessing the Client via HTTP Proxies.....	28
<b>9 Exceptions.....</b>	<b>30</b>
<b>A Change History.....</b>	<b>31</b>

# 1 SDK Introduction

## NLP Overview

Natural Language Processing (NLP) is a cloud service catered for enterprises and developers for text analysis and mining. It provides open APIs for efficient text processing.

## SDK Overview

The NLP software development kit (SDK) encapsulates the RESTful APIs provided by NLP to simplify application development. You can directly call API functions provided by NLP SDKs to use the service capabilities. Currently, the NLP SDK supports the Java version. For other languages, you can invoke corresponding services by calling APIs.

## Mappings Between Services and APIs

**Table 1-1** Mappings between services and APIs

Class	Method	API
NlpfClient	segment	POST /v1/{project_id}/nlp-fundamental/segment
	ner	POST /v1/{project_id}/nlp-fundamental/ner
	domainNer	POST /v1/{project_id}/nlp-fundamental/ner/domain
	getTextSimilarity	POST /v1/{project_id}/nlp-fundamental/text-similarity
	getSentenceVectors	POST /v1/{project_id}/nlp-fundamental/sentence-embedding
NlgClient	summary	POST /v1/{project_id}/nlg/summarization

Class	Method	API
NluClient	getSentiment	POST /v1/{project_id}/nlu/sentiment
	getDomainSentiment	POST /v1/{project_id}/nlu/sentiment/domain
	classifyText	POST /v1/{project_id}/nlu/classification
	getIntent	POST /v1/{project_id}/nlu/semantic-parser
MtClient	translateText	POST /v1/{project_id}/machine-translation/text-translation
	detectLanguage	POST /v1/{project_id}/machine-translation/language-detection

# 2 Authentication

---

Authentication is required to use service SDKs. Currently, authentication using AK/SK and the username and password is supported.

## AK/SK-based Authentication

You are advised to use AK/SK-based authentication, which requires you to obtain the AK and SK as follows.

1. Register with and log in to the HUAWEI CLOUD management console.
2. Hover the cursor on the username and select **My Credentials** from the drop-down list box.
3. Click the **Access Keys** tab and then click **Create Access Key**.
4. Enter the login password of the current user and the verification code received through email or SMS.
5. Click **OK** to download the AK/SK. Keep the AK/SK properly.

## Username and Password-based Authentication

You can use the HUAWEI CLOUD username and password to access APIs. Enter the username, password, and account in the constructor for creating the corresponding tool class to call the method, and access NLP.

# 3 Obtaining SDKs

**Table 3-1** provides details about how to obtain SDKs, regions and endpoints.

A region indicates the region where a service is located, and an endpoint indicates the endpoint region where a service is located. Only the region value is required in the SDK by default.

**Table 3-1** Obtaining SDKs and endpoints

SDK Name	Download Path	Obtain Regions and Endpoints
NLP SDKs	<a href="http://nlp-sdk.obs.cn-north-4.myhuaweicloud.com/nlp-client-1.0.1.jar">http://nlp-sdk.obs.cn-north-4.myhuaweicloud.com/nlp-client-1.0.1.jar</a>	<a href="#">Regions and Endpoints</a>

**Table 3-2** Supported regions and endpoints by NLP

Region	Endpoint Region	Endpoint	Protocol
CN North-Beijing4	cn-north-4	nlp-ext.cn-north-4.myhuaweicloud.com	HTTPS

# 4 Obtaining a Project ID

---

A project ID is required for calling NLP APIs. For details about how to obtain a project ID, see [Obtaining a Project ID](#).

# 5 Using the NLP Fundamentals SDK

## 5.1 Initializing the Client

The methods for initializing NLP SDK clients are the same. Construct the **AuthInfo** class and select an authentication method. There are two ways of authentication: AK/SK-based authentication and username and password-based authentication during initialization.

### AK/SK-based Authentication

```
//  
//Obtain the AK/SK, region, and project ID by referring to sections "Authentication", "Obtaining SDKs", and  
//Obtaining a Project ID".  
//For example, if the AK is {ak}, the SK is {sk}, the region is {region}, the project ID is {projectId}, the  
//client is constructed as follows:  
//  
AuthInfo authInfo = new AuthInfo("{ak}", "{sk}", "{region}", "{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.AKSK.  
//  
NlpfClient client = new NlpfClient(AuthMode.AKSK, authInfo);
```

### Username and Password-based Authentication

```
//  
//Obtain the region and project ID by referring to sections "Obtaining SDKs" and "Obtaining a Project ID".  
//For example, if the username is {userName}, the password is {pwd}, the account is {domainName}, the  
//region is {region}, the project ID is {projectId}, the client is constructed as follows:  
//An account can have two or more users. If a user uses the account for login, the value of userName  
//should be the same as that of domainName.  
//  
AuthInfo authInfo = new AuthInfo("{userName}", "{pwd}", "{domainName}", "{region}", "{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.TOKEN.  
//  
NlpfClient client = new NlpfClient(AuthMode.TOKEN, authInfo);
```

## 5.2 Calling NLP Fundamentals APIs

## 5.2.1 Word Segmentation

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object
creation.
NlpfClient client = new NlpfClient(AuthMode.AKSK,authInfo);

SegmentReq req = new SegmentReq();
//text: indicates text to be segmented, which is mandatory.
req.setText("Text to segment");

// Optional. SegmentConstant.POS_SWITCH_ON indicates enabling part-of-speech tagging and
SegmentConstant.POS_SWITCH_OFF indicates disabling part-of-speech tagging. The values are 1
and 0 respectively.
req.setPosSwitch(SegmentConstant.POS_SWITCH_ON);

//Optional. The default value is zh, indicating Chinese. Only Chinese is currently supported.
req.setLang("zh");

//Optional. Supported word segmentation criteria. Currently, Peking University standard (PKU) and
Chinese Penn Treebank (CTB) are supported. The default value is PKU.
req.setCriterion("PKU");

try {
    SegmentResp resp = client.segment(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The
client automatically processes exceptions.
}
```

- For details about **SegmentReq** parameters, see [Table 5-1](#). Obtain and set the parameters by using the getter and setter methods.

**Table 5-1 SegmentReq parameters**

Parameter	Mandato ry	Type	Description
text	Yes	String	Text to be segmented
posSwitch	No	Integer	1 indicates enabling part-of-speech tagging and 0 indicates disabling part-of-speech tagging. The default value is 0.
lang	No	String	Language. zh indicates Chinese. The default value is zh.
criterion	No	String	Supported word segmentation criteria. Currently, Peking University standard (PKU) and Chinese Penn Treebank (CTB) are supported. The default value is PKU.

- For details about **SegmentReq** parameters, see [Table 5-2](#). Obtain and set the parameters by using the getter and setter methods.

**Table 5-2 SegmentResp parameters**

Parameter	Type	Description
words	Array of Word	Word segmentation result

**Table 5-3 Data structure description of the Word field**

Parameter	Type	Description
content	String	Word text
pos	String	Part of speech of a word. For details, see <a href="#">Word Segmentation</a> .

## 5.2.2 NER (Basic Edition)

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NlpfClient client = new NlpfClient(AuthMode.AKSK,authInfo);
NerReq req = new NerReq();

//text: indicates text to be identified, which is mandatory.
req.setText("Text for analysis");

//Optional. zh indicates Chinese.
req.setLang("zh");

try {
    NerResp resp = client.ner(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details, see the section "Exceptions". The client automatically processes the exception.
}
```

- [Table 5-4](#) describes **NerReq** parameters.

**Table 5-4 NerReq parameters**

Parameter	Mandatory	Type	Description
text	Yes	String	Text to be identified
lang	No	String	Language to be identified. The default value is <b>zh</b> . Only Chinese is currently supported.

Parameter	Mandatory	Type	Description
domain	No	String	Supported domain type. Currently, only general domain is supported. The default value is <b>general</b> . This parameter is optional for APIs of the basic version.

- [Table 5-5](#) describes **NerResp** parameters.

**Table 5-5** NerResp parameters

Parameter	Type	Description
namedEntities	List<NamedEntity>	NER result list

- [Table 5-6](#) describes the data structure description of the **NamedEntity** field.

**Table 5-6** Data structure description of the **NamedEntity** field

Parameter	Type	Description
word	String	Entity text
tag	String	Entity and enumerated type. The options are as follows: <b>nr</b> (person name), <b>ns</b> (location), <b>nt</b> (organization), <b>tpt</b> (time point), <b>day</b> (date), <b>pct</b> (percentage), <b>mny</b> , (currency), <b>ord</b> (ordinal number), <b>qtt</b> (quantifier); <b>race</b> (nationality), <b>job</b> , and <b>email</b> .
offset	Integer	Starting position of the entity text in the text to be analyzed
len	Integer	Entity text length

### 5.2.3 NER (Domain-specific Edition)

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NlpfClient client = new NlpfClient(AuthMode.AKSK,authInfo);
NerReq req = new NerReq();

//text: indicates text to be identified, which is mandatory.
req.setText("Text for analysis");
```

```
//Optional. zh indicates Chinese.
req.setLang("zh");

try {
    NerResp resp = client.domainNer(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The
    client automatically processes exceptions.
}
```

- [Table 5-7](#) describes **NerReq** parameters.

**Table 5-7 NerReq** parameters

Parameter	Mandatory	Type	Description
text	Yes	String	Text to be identified
lang	No	String	Language to be identified. The default value is <b>zh</b> . Only Chinese is currently supported.
domain	No	String	Supported domain type. Currently, only general domain is supported. The default value is <b>general</b> . This parameter is mandatory for APIs of the domain-specific edition.

- [Table 5-8](#) describes **NerResp** parameters.

**Table 5-8 NerResp** parameters

Parameter	Type	Description
namedEntities	List<NamedEntity>	NER result list

- [Table 5-9](#) describes the data structure description of the **NamedEntity** field.

**Table 5-9** Data structure description of the **NamedEntity** field

Parameter	Type	Description
word	String	Entity text
tag	String	Entity and enumerated type. The options are as follows: <b>nr</b> (person name), <b>ns</b> (location), <b>nt</b> (organization), <b>tpt</b> (time point), <b>day</b> (date), <b>pct</b> (percentage), <b>mny</b> , (currency), <b>ord</b> (ordinal number), <b>qtt</b> (quantifier); <b>race</b> (nationality), <b>job</b> , and <b>email</b> .
offset	Integer	Starting position of the entity text in the text to be analyzed
len	Integer	Entity text length

## 5.2.4 Text Similarity

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NlpfClient client = new NlpfClient(AuthMode.AKSK,authInfo);

TextSimilarityReq req = new TextSimilarityReq();

//Two texts for similarity comparison
req.setText1("text-1");
req.setText2("text-2");

//Supported language of the text. The default value is zh, indicating Chinese. Only Chinese is currently supported.
req.setLang("zh");

try {
    TextSimilarityResp resp = client.getTextSimilarity(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The client automatically processes exceptions.
}
```

- **Table 5-10** describes **TextSimilarityReq** parameters.

**Table 5-10** **TextSimilarityReq** parameters

Parameter	Mandatory	Type	Description
text1	Yes	String	Text 1, on which the text similarity is to be computed. The text is encoded using UTF-8 and contains 1 to 512 characters.

Parameter	Mandatory	Type	Description
text2	Yes	String	Text 2, on which the text similarity is to be computed. The text is encoded using UTF-8 and contains 1 to 512 characters.
lang	No	String	Supported text language type. The default value is <b>zh</b> . Currently, only Chinese is supported.

- [Table 5-11](#) describes **TextSimilarityResp** parameters.

**Table 5-11** TextSimilarityResp parameters

Parameter	Type	Description
similarity	Double	Similarity score. The value ranges from 0 to 1. By default, eight digits are reserved after the decimal point.

## 5.2.5 Sentence Vector

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NlpfClient client = new NlpfClient(AuthMode.AKSK,authInfo);

SentenceVectorsReq req = new SentenceVectorsReq();

//Construct a sentence for which a sentence vector needs to be obtained.
List<String> sentences = new ArrayList<String>();
sentences.add("sentence-1");
sentences.add("sentence-2");
req.setSentences(sentences);

//Model name. Only general is supported.
req.setDomain("general");

try {
    SentenceVectorsResp resp = client.getSentenceVectors(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The client automatically processes exceptions.
}
```

- [Table 5-12](#) describes **SentenceVectorsReq** parameters.

**Table 5-12 SentenceVectorsReq** parameters

Parameter	Mandatory	Type	Description
sentences	Yes	List<String>	Text list. The text is encoded using UTF-8 and contains 1 to 512 characters. The text list contains 1 to 1,000 text data records.
domain	No	String	Name of the model used for computing the sentence vector. Currently, only the default value <b>general</b> is available.

- [Table 5-13](#) describes **SentenceVectorsResp** parameters.

**Table 5-13 SentenceVectorsResp** parameters

Parameter	Type	Description
vectors	List<List<Double>>	Sentence vector result list. Sentence vectors are returned according to the input sentence sequence.

## 5.3 Accessing the Client via HTTP Proxies

You can access all clients of the NLP SDK via HTTP proxies.

- Example

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NlpfClient client = new NlpfClient(AuthMode.AKSK,authInfo);
```

```
//Enable the proxy that does not require authentication. The first parameter indicates the proxy IP address,
and the second parameter indicates the proxy port number.
client.setProxy("127.0.0.1",8888);
```

```
//Enable the proxy that requires authentication of the username and password. username indicates the
username, and password indicates the password.
client.setProxy("127.0.0.1",8888,"username","password");
```

# 6 Using the LU SDK

## 6.1 Initializing the Client

### AK/SK-based Authentication

```
//  
//Obtain the AK/SK, region, and project ID by referring to sections "Authentication", "Obtaining SDKs", and  
//Obtaining a Project ID".  
//For example, if the AK is {ak}, the SK is {sk}, the region is {region}, the project ID is {projectId}, the  
//client is constructed as follows:  
//  
AuthInfo authInfo = new AuthInfo("{ak}","{sk}","{region}","{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.AKSK.  
//  
NluClient client = new NluClient(AuthMode.AKSK,authInfo);
```

### Username and Password-based Authentication

```
//  
//Obtain the region and project ID by referring to sections "Obtaining SDKs" and "Obtaining a Project ID".  
//For example, if the username is {userName}, the password is {pwd}, the account is {domainName}, the  
//region is {region}, the project ID is {projectId}, the client is constructed as follows:  
//An account can have two or more users. If a user uses the account for login, the value of userName  
//should be the same as that of domainName.  
//  
AuthInfo authInfo = new AuthInfo("{userName}","{pwd}","{domainName}","{region}","{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.TOKEN.  
//  
NluClient client = new NluClient(AuthMode.TOKEN,authInfo);
```

## 6.2 Calling LU APIs

### 6.2.1 Sentiment Analysis (Basic Edition)

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object
creation.
NluClient client = new NluClient (AuthMode.AKSK,authInfo);

GeneralSentimentReq req = new GeneralSentimentReq();
req.setContent("Text for analysis");

try {
    GeneralSentimentResp resp = client.getSentiment(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The
client automatically processes exceptions.
}
```

- **Table 6-1** describes **GeneralSentimentReq** parameters.

**Table 6-1** GeneralSentimentReq parameters

Parameter	Mandatory	Type	Description
content	Yes	String	Text to be analyzed. The text must be encoded using UTF-8. The value contains a maximum of 400 characters. If the value exceeds 400 characters, only the first 400 characters are detected.

- **Table 6-2** describes **GeneralSentimentResp** parameters.

**Table 6-2** GeneralSentimentResp parameters

Parameter	Type	Description
result	Result	Encapsulated object of the returned result

- **Table 6-3** describes the data structure description of the **Result** field.

**Table 6-3** Data structure description of the **Result** field

Parameter	Type	Description
content	String	Text to be analyzed.
label	Integer	Positive and negative labels are included. <ul style="list-style-type: none"> <li>• <b>1</b>: positive</li> <li>• <b>0</b>: negative</li> </ul>
confidence	Float	Confidence level of a label

## 6.2.2 Sentiment Analysis (Domain-specific Edition)

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object
creation.
NluClient client = new NluClient (AuthMode.AKSK,authInfo);

GeneralSentimentReq req = new GeneralSentimentReq();
req.setContent("Text for analysis");

//Domain type. SentimentDomainConstant.DOMAIN_TYPE_CAR indicates the automotive domain.
The value is 2.
//SentimentDomainConstant.DOMAIN_TYPE_E_COMMERCE indicates the e-commerce domain. The
value is 1.
req.setType(SentimentDomainConstant.DOMAIN_TYPE_CAR);

try {
    GeneralSentimentResp resp = client.getDomainSentiment(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The
client automatically processes exceptions.
}
```

- **Table 6-4** describes **GeneralSentimentReq** parameters.

**Table 6-4** GeneralSentimentReq parameters

Parameter	Mandatory	Type	Description
content	Yes	String	Text to be analyzed. The text must be encoded using UTF-8. Only Chinese text sentiment analysis is supported. If <b>type</b> is set to <b>1</b> (e-commerce), the value contains a maximum of 200 characters. If the value exceeds 200 characters, only the first 200 characters are detected. If <b>type</b> is set to <b>2</b> (automotive), the value contains a maximum of 400 characters. If the value exceeds 400 characters, only the first 400 characters are detected.
type	No	Integer	The options are as follows: <b>1</b> : E-commerce comments <b>2</b> : Automotive comments

- **Table 6-5** describes **GeneralSentimentResp** parameters.

**Table 6-5** GeneralSentimentResp parameters

Parameter	Type	Description
result	Result	Encapsulated object of the returned result

- **Table 6-6** describes the data structure description of the **Result** field.

**Table 6-6** Data structure description of the **Result** field

Parameter	Type	Description
content	String	Text to be analyzed.
label	Integer	Positive and negative labels are included. <ul style="list-style-type: none"> <li>• <b>1</b>: positive</li> <li>• <b>0</b>: negative</li> </ul>
confidence	Float	Confidence level of a label

## 6.2.3 Text Classification

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NluClient client = new NluClient (AuthMode.AKSK,authInfo);

ClassifyReq req = new ClassifyReq();
req.setContent("Text for classification");

//Classified domain type. ClassifyDomainConstant.DOMAIN_TYPE_AD indicates advertisement and its value is 1. Currently, only the advertisement type is supported.
req.setDomain(ClassifyDomainConstant.DOMAIN_TYPE_AD);

try {
    ClassifyResp resp = client.classifyText(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The client automatically processes exceptions.
}
```

- [Table 6-7](#) describes **ClassifyReq** parameters.

**Table 6-7** **ClassifyReq** parameters

Parameter	Mandatory	Type	Description
content	Yes	String	Text to be analyzed. The text is encoded using UTF-8 and the value contains a maximum of 400 characters. If the value exceeds 400 characters, only the first 400 characters are detected.
domain	No	Integer	The default value is <b>1</b> . Currently, only advertisement detection is supported. <b>1</b> : Advertisement detection

- [Table 6-8](#) describes **ClassifyResp** parameters.

**Table 6-8 ClassifyResp parameters**

Parameter	Type	Description
result	Result	Encapsulated object of the returned result

- **Table 6-9** describes the data structure description of the **Result** field.

**Table 6-9** Data structure description of the **Result** field

Parameter	Type	Description
content	String	Text to be analyzed.
label	Integer	Positive and negative labels are included. <ul style="list-style-type: none"> <li>• <b>1</b>: advertisement</li> <li>• <b>0</b>: non-advertisement</li> </ul>
confidence	Float	Confidence level of a label

## 6.2.4 Intent Understanding

- **Example request**  

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NluClient client = new NluClient (AuthMode.AKSK,authInfo);

IntentReq req = new IntentReq();
req.setText("Text for analysis");

//Supported language of the text. zh indicates Chinese. Only Chinese is currently supported.
req.setLang("zh");

try {
    IntentResp resp = client.getIntent(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The client automatically processes exceptions.
}
```
- **Table 6-10** describes **IntentReq** parameters.

**Table 6-10 IntentReq parameters**

Parameter	Mandatory	Type	Description
text	Yes	String	Text to be analyzed. The text is encoded using UTF-8 and the value contains a maximum of 400 characters. If the value exceeds 400 characters, only the first 400 characters are detected.

Parameter	Mandatory	Type	Description
lang	No	String	Supported language type. Currently, only Chinese is supported. The default value is <b>zh</b> .

- [Table 6-11](#) describes **IntentResp** parameters.

**Table 6-11** IntentResp parameters

Parameter	Type	Description
result	IntentResult	Encapsulated object of the returned result

- [Table 6-12](#) describes the data structure description of the **IntentResult** field.

**Table 6-12** Data structure description of the **IntentResult** field

Parameter	Type	Description
text	String	Text to be analyzed.
label	String	Intent label of the text to be analyzed. Labels are classified into the following types: weather, time, news, joke, translation, notification, alarm, music, and others.
confidence	Float	Confidence level of a label
slots	List of Slot	Entity text list of the sentence to be analyzed. For details about the slot data structure, see <a href="#">Table 6-13</a> .

**Table 6-13** Data structure description of the **Slot** field

Parameter	Type	Description
word	String	Entity text to be analyzed

Parameter	Type	Description
tag	String	Entity type. Entity types supported by each intent category are: <b>weather</b> : date, time, and location <b>time</b> : location and timezone <b>news</b> : genre <b>joke</b> : genre <b>translation</b> : content <b>notification</b> : content, date, time, and singer <b>alarm</b> : date and time <b>music</b> : singer, song, and content
offset	Integer	Starting position of the entity text in the text to be analyzed
length	Integer	Entity text length
normalizedWord	String	Synonym or other normalized word. The default value is the original <b>word</b> text.

### 6.3 Accessing the Client via HTTP Proxies

You can access all clients of the NLP SDK via HTTP proxies.

- Example

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NluClient client = new NluClient(AuthMode.AKSK,authInfo);

//Enable the proxy that does not require authentication. The first parameter indicates the proxy IP address,
and the second parameter indicates the proxy port number.
client.setProxy("127.0.0.1",8888);

//Enable the proxy that requires authentication of the username and password. username indicates the
username, and password indicates the password.
client.setProxy("127.0.0.1",8888,"username","password");
```

# 7 Using the LG SDK

## 7.1 Initializing the Client

### AK/SK-based Authentication

```
//  
//Obtain the AK/SK, region, and project ID by referring to sections "Authentication", "Obtaining SDKs", and  
//Obtaining a Project ID".  
//For example, if the AK is {ak}, the SK is {sk}, the region is {region}, the project ID is {projectId}, the  
//client is constructed as follows:  
//  
AuthInfo authInfo = new AuthInfo("{ak}","{sk}","{region}","{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.AKSK.  
//  
NlgClient client = new NlgClient(AuthMode.AKSK,authInfo);
```

### Username and Password-based Authentication

```
//  
//Obtain the region and project ID by referring to sections "Obtaining SDKs" and "Obtaining a Project ID".  
//For example, if the username is {userName}, the password is {pwd}, the account is {domainName}, the  
//region is {region}, the project ID is {projectId}, the client is constructed as follows:  
//An account can have two or more users. If a user uses the account for login, the value of userName  
//should be the same as that of domainName.  
//  
AuthInfo authInfo = new AuthInfo("{userName}","{pwd}","{domainName}","{region}","{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.TOKEN.  
//  
NlgClient client = new NlgClient(AuthMode.TOKEN,authInfo);
```

## 7.2 Calling LG APIs

### 7.2.1 Text Summarization

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object
creation.
NlgClient client = new NlgClient(AuthMode.AKSK,authInfo);

SummaryReq req = new SummaryReq();

//Text content
req.setContent("Almost 600 plant species have been wiped from the planet in the past 250 years," +
    "more than twice the number of birds," +
    "mammal and amphibian species that have met the same fate," +
    "according to a new study." +
    "The research -- published Monday in Nature, Ecology & Evolution journal --" +
    "found that 571 plant species have disappeared from the wild worldwide," +
    "and that plant extinction is occurring up to 500 times faster than the rate it would without
human intervention." +
    "For comparison," +
    "the researchers said animal extinction is occurring at least 1,000 times faster than the
normal rate of extinction," +
    "however the report notes that researchers believe the plant extinction rate has been
underestimated." +
    "The Royal Botanic Gardens, " +
    "Kew and Stockholm University researchers say it's the first time scientists have compiled a
global overview of which plants have already become extinct." +
    "Most people can name a mammal or bird that has become extinct in recent centuries," +
    "but few can name an extinct plant," +
    "report co-author Aleys Humphreys said."+
    "Plants on islands," +
    "in the tropics and in Mediterranean climates had the highest rates of plant extinction," +
    "as these were areas home to unique species vulnerable to human activities." " +
    "The study said the increase in the plant extinction rate could be due to the loss of habitat of
species located in a small geographic area." +
    "Some examples of plants that have gone extinct in the past two centuries are the aromatic
Chile sandalwood and the banded trinity," +
    "which has no leaves -" +
    "only its flowers are visible above the ground. ");

//Optional. Text title
req.setTitle("Almost 600 plant species have become extinct in the last 250 years");

//Optional. Length limit of the generated text summarization. If the value is between 0 and 1, the
length percentage is close to the value. If the value is greater than 1, the summarization is the one
whose text length is closest to the value.
req.setLengthLimit(0.3f);

//Optional. The default value is zh, indicating Chinese. Only Chinese is currently supported.
req.setLang("zh");

try {
    SummaryResp resp = client.summary(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The
client automatically processes exceptions.
}
```

- [Table 7-1](#) describes **SummaryReq** parameters.

**Table 7-1 SummaryReq** parameters

Parameter	Mandatory	Type	Description
content	Yes	String	Text body. Currently, the title is encoded using UTF-8. The text length ranges from 1 to 10,000 characters. (Note: The text language type must be consistent with the value of <b>lang</b> .)
title	No	String	Text title. Currently, the title is encoded using UTF-8. The text length ranges from 0 to 1,000 characters.
lang	No	String	Supported text language type. Currently, Chinese ( <b>zh</b> ) and English ( <b>en</b> ) are supported. The default value is <b>zh</b> .
lengthLimit	No	Float	Length limit of the generated summary. <ul style="list-style-type: none"> <li>If the value of <b>length_limit</b> is greater than <b>1</b>, the returned result is the summary whose length is the closest to the value.</li> <li>If the value of <b>length_limit</b> ranges from <b>0</b> to <b>1</b>, the returned result is the summary whose length percentage is closest to the value.</li> </ul> The default value is <b>0.3</b> .

- [Table 7-2](#) describes **SummaryResp** parameters.

**Table 7-2 SummaryResp** parameters

Parameter	Type	Description
summary	String	Summary result returned based on the text request body

## 7.3 Accessing the Client via HTTP Proxies

You can access all clients of the NLP SDK via HTTP proxies.

- Example

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
NlgClient client = new NlgClient(AuthMode.AKSK,authInfo);

//Enable the proxy that does not require authentication. The first parameter indicates the proxy IP address,
and the second parameter indicates the proxy port number.
client.setProxy("127.0.0.1",8888);

//Enable the proxy that requires authentication of the username and password. username indicates the
```

username, and *password* indicates the password.  
`client.setProxy("127.0.0.1",8888,"username","password");`

# 8 Using the MT SDK

## 8.1 Initializing the Client

### AK/SK-based Authentication

```
//  
//Obtain the AK/SK, region, and project ID by referring to sections "Authentication", "Obtaining SDKs", and  
//Obtaining a Project ID".  
//For example, if the AK is {ak}, the SK is {sk}, the region is {region}, the project ID is {projectId}, the  
//client is constructed as follows:  
//  
AuthInfo authInfo = new AuthInfo("{ak}","{sk}","{region}","{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.AKSK.  
//  
MtClient client = new MtClient(AuthMode.AKSK,authInfo);
```

### Username and Password-based Authentication

```
//  
//Obtain the region and project ID by referring to sections "Obtaining SDKs" and "Obtaining a Project ID".  
//For example, if the username is {userName}, the password is {pwd}, the account is {domainName}, the  
//region is {region}, the project ID is {projectId}, the client is constructed as follows:  
//An account can have two or more users. If a user uses the account for login, the value of userName  
//should be the same as that of domainName.  
//  
AuthInfo authInfo = new AuthInfo("{userName}","{pwd}","{domainName}","{region}","{projectId}");  
  
//  
//After AuthInfo is constructed, initialize the corresponding client. In this case, set the first parameter of the  
//construction method to AuthMode.TOKEN.  
//  
MtClient client = new MtClient(AuthMode.TOKEN,authInfo);
```

## 8.2 Calling MT APIs

### 8.2.1 Text Translation

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object
creation.
MtClient client = new MtClient(AuthMode.AKSK,authInfo);

TextTranslateReq req = new TextTranslateReq();

//Text to be translated
req.setText("Text to translate");

//Source language. zh indicates Chinese and auto indicates automatic identification.
req.setFrom("zh");

//Target language. en indicates English.
req.setTo("en");

//Optional. Application scenario. Currently, only common is supported.
req.setScene("common");

try {
    TextTranslateResp resp = client.translateText(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The
client automatically processes exceptions.
}
```

- [Table 8-1](#) describes **TextTranslateReq** parameters.

**Table 8-1** TextTranslateReq parameters

Parameter	Mandatory	Type	Description
text	Yes	String	Text to be translated
from	Yes	String	Abbreviation of the source language. <b>auto</b> indicates automatic identification. <a href="#">Table 8-3</a> lists the supported languages.
to	Yes	String	Abbreviation of the target language. For details about supported languages, see <a href="#">Table 8-3</a> .
scene	No	String	Only the default value <b>common</b> is currently supported, indicating a common scenario.

- [Table 8-2](#) describes **TextTranslateResp** parameters.

**Table 8-2** TextTranslateResp parameters

Parameter	Type	Description
srcText	String	Text to be translated
translatedText	String	Translation result

Parameter	Type	Description
from	String	Source language when the API call succeeds (if the source language is set to <b>auto</b> , the language detection result is displayed). It is encoded using UTF-8. This parameter is not included when the API call fails.
to	String	Target language when the API call succeeds. It is encoded using UTF-8. This parameter is not included when the API call fails.

**Table 8-3** Supported languages

Language (from)	Language (to)	Description
zh	en	Translation from Chinese to English
en	zh	Translation from English to Chinese
auto	en	Translation from Chinese to English
auto	zh	Translation from English to Chinese

## 8.2.2 Language Detection

- Example request

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
MtClient client = new MtClient(AuthMode.AKSK,authInfo);

LanguageDetectionReq req = new LanguageDetectionReq();

//Content to be detected
req.setText("Play the Qinghuaci of Jay Chou");

try {
    LanguageDetectionResp resp = client.detectLanguage(req);
} catch (NlpException e) {
    //A failure is thrown as an exception. For details about exceptions, see section "Exceptions". The client automatically processes exceptions.
}
```

- [Table 8-4](#) describes **LanguageDetectionReq** parameters.

**Table 8-4 LanguageDetectionReq parameters**

Parameter	Mandatory	Type	Description
text	Yes	String	Text to be detected

- [Table 8-5](#) describes **LanguageDetectionResp** parameters.

**Table 8-5 LanguageDetectionResp parameters**

Parameter	Type	Description
detectedLanguage	String	Abbreviation of the detected language. For details about supported languages, see <a href="#">Table 8-6</a> .

**Table 8-6 Supported languages**

Language	Abbreviation
Chinese	zh
English	en
Russian	ru
Japanese	ja
German	de
French	fr
Spanish	es
Portuguese	pt
Italian	it
Turkish	tr
Arabic	ar

## 8.3 Accessing the Client via HTTP Proxies

You can access all clients of the NLP SDK via HTTP proxies.

- Example

```
//You are advised to construct and use the client in a singleton pattern to avoid frequent object creation.
MtClient client = new MtClient(AuthMode.AKSK,authInfo);

//Enable the proxy that does not require authentication. The first parameter indicates the proxy IP address,
and the second parameter indicates the proxy port number.
client.setProxy("127.0.0.1",8888);
```

```
//Enable the proxy that requires authentication of the username and password. username indicates the  
username, and password indicates the password.  
client.setProxy("127.0.0.1",8888,"username","password");
```

# 9 Exceptions

If the NLP API fails to be called, the **NlpException** class is thrown. You can locate the cause of the problem based on exceptions.

**Table 9-1** NlpException field description

Parameter	Type	Mandatory	Description
message	String	Yes	Error information overview, which is inherited from the <b>Exception</b> class.
responseCode	Integer	No	HTTP error code returned by the server. For details, see <a href="#">Status Code</a> .
reponseBody	String	No	Error body returned by the server, including the <b>errorCode</b> and <b>errorMessage</b> fields. For details, see <a href="#">Error Code</a> .

For details about status codes, see [Status Code](#) in *API Reference*. For details about error codes, see [Error Code](#) in *API Reference*.

# A Change History

---

Released On	What's New
2019-12-27	This is the first official release.