pitney bowes

Spectrum Technology Platform

Version 0-SNAPSHOT

Information Extraction-Handbuch

Inhalt

1 - Einführung

2 - Entitätsextraktion

3 - Beziehungsextraktion

4 - Textkategorisierung

5 - Schrittreferenz

1 - Einführung

In this section

Information Extraction-Modul von

Das Information Extraction-Modul von bietet erweiterte Textverarbeitungsfunktionen und Informationsextraktion aus natürlichsprachigem Eingabetext.

Es verfügt über vortrainierte Modelle, die dazu verwendet werden, Entitäten aus einem Eingabetext zu extrahieren, Beziehungen zwischen Entitäten zu ermitteln und die Kategorie zuzuweisen, zu der der Text gehört.

Bereitgestellte Funktionen

Entitätsextraktion	Extrahiert Entitäten aus unstrukturierten Daten und klassifiziert sie in Typen wie **Ort**, **Datum**, **Organisation**, **Eigennamen**, **Adresse** und **Person**. Das Modul wird mit einigen *vorhandenen Entitäten* bereitgestellt. Es ist jedoch auch in der Lage, Modelle basierend auf Ihren Anforderungen zu trainieren. Weitere Informationen zum Trainieren eines Modells und zum Definieren benutzerdefinierter Entitäten finden Sie unter **Benutzerdefinierte Entitäten** auf Seite 8.
Beziehungsextraktion	Identifiziert den Beziehungstyp, der die Entitäten in einem natürlichsprachigen Eingabetext verbindet.
Textkategorisierung	Ordnet Ihrem unstrukturierten Text basierend auf seinem Inhalt Kategorien wie E-Mail, medizinische Berichte und Sport zu. Vor der Kategorisierung müssen Sie ein *Textkategorisierungsmodell* mithilfe der Administrationsumgebung trainieren. Diese Funktion kann verwendet werden, um Gesundheitsversorgungsberichte von Patienten zu indizieren, Dokumente nach Domänen und Unterdomänen zu klassifizieren und E-Mails in SPAM und Nicht-SPAM zu kategorisieren, neben anderen Anwendungen. Sie klassifiziert auch die identifizierten Kategorien, basierend auf dem Ausmaß, in dem Ihr Text mit diesen übereinstimmt.

Unterstützte Sprachen

Alle Schritte des Information Extraction-Moduls des aktuellen Releases unterstützen Information Extraction-Funktionen nur für Eingabetext in *Englisch*.

Anmerkung: Der **Entity Extractor**-Schritt unterstützt zusätzlich zu Englisch folgende Sprachen in einer *Beta-Phase*:

es	Spanisch (Mexiko)
fr	Französisch
de	Deutsch

pt	Portugiesisch (Brasilien)

Wichtig: Diese Sprachen in der *Beta*-Phase stehen nur für *Custom Entity* und nicht für vorher vorhandene Entitäten zur Verfügung.

Modellsicherheit

Um mit Information Extraction verschiedene Funktionen ausführen zu können, müssen auf der **Management Console** Sicherheitsberechtigungen zugewiesen werden:

- Berechtigungen zum Anzeigen sind erforderlich, um das Modell zu kategorisieren oder aufzulisten.
- Berechtigungen zum Ändern sind erforderlich, um das Modell (wenn es bereits vorhanden ist) erneut zu trainieren oder zu importieren.
- Berechtigungen zum Erstellen sind erforderlich, um das Modell zu importieren oder erneut zu trainieren.
- Berechtigungen zum Löschen sind erforderlich, um das Modell zu löschen.

2 - Entitätsextraktion

In this section

Entitätsextraktor

Entitätsextraktion ist der Prozess, bei dem Entitäten in unstrukturierten Daten identifiziert und aus diesen extrahiert werden. Sie können die vorher vorhandenen Entitäten verwenden, die mit dem **Entity Extractor**-Schritt geliefert werden, oder ein Modell trainieren und benutzerdefinierte Entitäten extrahieren. Weitere Informationen zum Trainieren eines Modells und zum Definieren benutzerdefinierter Entitäten finden Sie unter **Benutzerdefinierte Entitäten** auf Seite 8.

Vorher vorhandene Entitäten

Vorher vorhandene Entitäten sind Entitäten, die im **Information Extraction**-Modul enthalten sind.

Um eine Liste mit den vorher vorhandenen Entitäten anzuzeigen, öffnen Sie den **Entity Extractor**-Schritt, aktivieren Sie das Kästchen **System-Standardoptionen mit den folgenden Werten überschreiben** und klicken Sie auf **Schnell hinzufügen**. Die Liste mit den Entitäten wird im Bereich **Entität auswählen** angezeigt.

- *Person*
- *Address*
- *ProperNouns*
- *ISBN*
- *CreditCard*
- *ZipCode*
- *WebAddress*
- *Mention*
- *HashTag*
- *SSN*
- *Phone*
- *Email*
- *Date*
- *Location*
- *Organization*

Befolgen Sie die restlichen Schritte in diesem Kapitel, um diese Entitäten aus Ihren Daten zu extrahieren.

Extrahieren vorher vorhandener Entitäten

1. Erstellen Sie einen Datenfluss, der einen **Read from Documents**-Quellschritt, einen **Entity Extractor**-Schritt und einen Zielschritt wie **Write to File** oder **Write to XML** umfasst.
2. Verweisen Sie im Quellschritt auf Ihre Eingabedatei.
3. Wählen Sie im **Entity Extractor**-Schritt die Entitäten basierend auf den Daten, die Sie aus der Eingabedatei extrahieren möchten, aus. Wenn Sie beispielsweise die Namen aller Personen und alle Adressen in der Datei auswählen möchten, wählen Sie die Entitäten *Address* und *Person* aus.

   Anmerkung: Bei *Address* und *Person* handelt es sich um die Standardentitäten. Um die Daten basierend auf einer anderen Entität zu extrahieren, aktivieren Sie das Kästchen **System-Standardoptionen mit den folgenden Werten überschreiben** und klicken Sie auf **Schnell hinzufügen**. Die Liste mit den Entitäten wird im Bereich **Entität auswählen** angezeigt.

4. Aktivieren Sie das Kontrollkästchen **Anzahl der Ausgabeentitäten**, um die Häufigkeit der Daten in der Eingabedatei bezüglich der angegebenen Entitäten abzurufen.
5. Klicken Sie auf **OK**.
6. Führen Sie den Auftrag aus.

Benutzerdefinierte Entitäten

Sie können Modelle ähnlich wie vorhandene Entitäten auch trainieren, um benutzerdefinierte Entitäten abzurufen. Diese Entitäten können zu einer beliebigen Domäne gehören und eines beliebigen Typs sein. Sie können beispielsweise medizinische Texte verwenden, um eine Liste von Diagnosen oder Medikamenten zu extrahieren. Das Verfahren, benutzerdefinierte Entitäten zu extrahieren, umfasst die folgenden Schritte:

1. Vorbereiten der Daten: Vorbereiten der Eingabe- und der Testdatei
2. Konfigurieren der Optionen: Erstellen einer Datei mit Trainingsoptionen, die Informationen zum Modell und die beim Training des Modells anzuwendenden Optionen enthält
3. Trainieren des Modells
4. Extrahieren der Entitäten

Wenn Sie alle Schritte erfolgreich durchführen, wird der neue Entitätstyp zur Liste im **Entity Extractor**-Schritt hinzugefügt und Sie können ihn verwenden, um aus unstrukturierten Daten Details zu extrahieren.

Vorbereiten der Daten für benutzerdefinierte Entitäten

Der erste Schritt beim Erstellen von benutzerdefinierten Entitäten ist die Vorbereitung Ihrer Eingabedatei und Ihrer Testdatei. Für das Feature „Benutzerdefinierte Entitäten“ müssen die Entitäten in diesen Dateien von magicWord umgeben sein, das Sie in Ihrer Datei mit Trainingsoptionen angeben. (Dies wird im nächsten Thema behandelt.)

Angenommen Sie möchten aus den unstrukturierten Daten in Ihrer Eingabedatei Diagnosen extrahieren und haben in Ihrer Datei mit Trainingsoptionen das magicWord *DIAGNOSIS* angegeben. Jedes Mal, wenn die Bezeichnung einer Krankheit oder eines Leidens im Text erscheint, wird das Wort wie folgt von diesem magicWord umschlossen:

```
The term diagnostic criteria designates the specific combination of
signs, symptoms, and test results that the clinician uses to attempt
to determine the correct diagnosis. Some examples of diagnostic
criteria, also known as clinical case definitions, are: Amsterdam
criteria for DIAGNOSIShereditary nonpolyposis colorectal cancerDIAGNOSIS
McDonald criteria for DIAGNOSISmultiple sclerosisDIAGNOSIS ACR criteria

for DIAGNOSISsystemic lupus erythematosusDIAGNOSIS Centor criteria for

DIAGNOSISstrep throatDIAGNOSIS.
```

Informationen zum Identifizieren des magicWord finden Sie im nächsten Thema.

Konfigurieren der Optionen für benutzerdefinierte Entitäten

Dies beinhaltet die Erstellung einer Datei mit `Trainingsoptionen`, die Informationen zu Ihrem Modell und die beim Training des Modells anzuwendenden Optionen enthält. Diese Datei muss im XML-Format mit UFT-8-Codierung vorliegen. Sie muss folgende Features für Header und das erforderliche Training enthalten:

Header in der Datei mit Trainingsoptionen

Der Header enthält Details zu dem Modell, dem Testpfad und Eingabedateien sowie zum Schlüsselwort für Anmerkungen zu benutzerdefinierten Entitäten.

- `modelName`: Name des benutzerdefinierten Modells
- `modelType`: Der Typ des benutzerdefinierten Modells (der *CustomEntity* lautet).
- `modelDescription`: Beschreibung des benutzerdefinierten Modells
- `inputFilePath`: Pfad der markierten Datei, die zum Trainieren des Modells verwendet wird (Eingabedatei)
- `testFilePath`: Pfad der Datei, die zum Testen des Modells verwendet wird
- `magicWord`: Schlüsselwort für Anmerkungen zu benutzerdefinierten Entitäten

- `language`: Die im Text verwendete Sprache.

  Anmerkung: Englisch wird unterstützt. Niederländisch, Französisch, Deutsch und Spanisch befinden sich in der Beta-Phase.

Trainingsfeatures

Sie können die benutzerdefinierten Entitäten mithilfe der folgenden Trainingsfeatures erstellen.

- **Sprachliche Features**: Für die Angabe der Spracheigenschaften
  - `POSTagger`: Markieren zum Identifizieren von Wortarten, wie z. B. Nomen, Pronomen, Adjektiven und Verben.

```
<trainingFeature>
   <featureName>POSTagger</featureName>
</trainingFeature>
```

- **Orthografische Features**: Für die Angabe der strukturellen Eigenschaften
  - `CaseIdentifier`: Gibt an, ob die benutzerdefinierten Entitäten in Großbuchstaben, in Kleinbuchstaben oder in einer Mischung aus beidem geschrieben werden.

```
<trainingFeature>
 <featureName>CaseIdentifier</featureName>
</trainingFeature>
```

  - `NumericIdentifier`: Gibt an, ob die benutzerdefinierten Entitäten numerisch oder alphanumerisch sind.

```
<trainingFeature>
 <featureName>NumericIdentifier</featureName>
</trainingFeature>
```

  - `1st2ndIdentifier`: Gibt an, ob es sich bei den benutzerdefinierten Entitäten um Ordnungszahlen wie 1., 2. und 3. handelt.

```
<trainingFeature>
 <featureName>1st2ndIdentifier</featureName>
</trainingFeature>
```

  - `PatternMatcher`: Vergleicht Wörter mithilfe von regulären Ausdrücken mit mindestens einem Muster. Wenn mehrere Ausdrücke angegeben sind, wird die Join-Bedingung `AND` für alle Ausdrücke oder `OR` (Standard) für einen beliebigen Ausdruck verwendet.

```
<trainingFeature>
 <featureName>PatternMatcher</featureName>
  <featureParams>
   <entry>
```

```
    <key>RegEx1</key>
    <value>b[aeiou]t</value>
   </entry>
   <entry>
    <key>RegEx2</key>
    <value>b[xyz]t</value>
   </entry>
   <entry>
    <key>JoinCondition</key>
    <value>AND</value>
   </entry>
  </featureParams>
</trainingFeature>
```

- **Schlüsselwortfeatures**: Zum Definieren der Liste mit Schlüsselwörtern
  - `CategoryKeywords`: Gibt eine Kategorie für eine Liste mit Schlüsselwörtern an, die zu mehreren benutzerdefinierten Listen gehören. Beispiel: „Wochentage“ in der Liste `CategoryKeywords` enthält die `Schlüsselwörter` Montag, Dienstag, Mittwoch, Donnerstag und Freitag.

    Dieses Feature kann optional angeben, ob beim Abgleich die Groß-/Kleinschreibung beachtet werden soll. Bei einer Verwendung lautet der Standard `true`.

```
<trainingFeature>
 <featureName>CategoryKeywords</featureName>
 <featureParams>
  <entry>
   <key>Weekdays</key>
                    <!-- List of weekdays -->
   <value>Monday,Tuesday,Wednesday,Thursday,Friday</value>
  </entry>
  <entry>
   <key>WeekendDays</key>
                    <!-- List of weekend days -->
   <value>Saturday,Sunday</value>
  </entry>
  <entry>
   <key>CaseSensitive</key>
                    <value>True</value>
  </entry>
 </featureParams>
</trainingFeature>
```

  - `KeyWords`: Sucht nach Wörtern, die Sie als zu einer benutzerdefinierten Liste gehörig angegeben haben, z. B. *DaysOfWeek* oder *Month*. Gibt zudem optional an, ob beim Abgleich die Groß-/Kleinschreibung beachtet werden soll. Bei einer Verwendung lautet der Standard „`true`“.

```
<trainingFeature>
 <featureName>KeyWords</featureName>
 <featureParams>
```

```
  <entry>
   <key>KeyWordList</key>
   <value>Monday,Tuesday</value>
  </entry>
  <entry>
   <key>CaseSensitive</key>
   <value>False</value>
  </entry>
 </featureParams>
</trainingFeature>
```

- `Substring`: Extrahiert eine Teilzeichenfolge wie in den Parametern angegeben. Kann auch zum Extrahieren von Präfixen und Suffixen verwendet werden.
  - `StartLocation`: Links oder rechts. Position, an der die Teilzeichenfolge extrahiert werden soll. Der Standard ist `Links`.
  - `StartPosition`: Startposition der Teilzeichenfolge. Der Standardwert ist `0`.
  - `EndPosition`: Endposition der Teilzeichenfolge. Der Standardwert ist `3`.
  - `MinLength`: Minimale Länge des Worts, auf das dieses Feature angewendet werden soll. Der Standardwert ist `3`.

```
<trainingFeature>
 <featureName>Substring</featureName>
  <featureParams>
   <entry>
    <key>StartLocation</key>
   </entry>
   <entry>
    <key>StartPosition</key>
    <value>1</value>
   </entry>
   <entry>
    <key>EndPosition</key>
    <value>4</value>
   </entry>
   <entry>
    <key>MinLength</key>
  </featureParams>
</trainingFeature>
```

- **Lexikalische Features**: Für die Angabe der Eigenschaften von Lexemen
  - `FeatureWindow`: Gibt das Fenster für die Featuregenerierung an

```
<trainingFeature>
 <featureName>FeatureWindow</featureName>
 <!-- Number of preceding tokens used to create the feature set.
Default is 3 -->
  <entry>
   <key>Before</key>
```

```
    <value>1</value>
   </entry>
  <!-- Number of succeeding tokens used to create the feature set.
 Default is 3 -->
   <entry>
    <key>After</key>
    <value>2</value>
   </entry>
 </trainingFeature>
```

Unten finden Sie eine vollständige Beispieldatei mit Trainingsoptionen für benutzerdefinierte Entitäten:

```
<trainingOptions>
 <modelName>CustomModel</modelName>
 <modelType>CustomEntity</modelType>
 <modelDescription>CustomDiagnosesModel</modelDescription>

<inputFilePath>C:/SpectrumIE/custom_model/Custom_Input.csv</inputFilePath>

 <testFilePath>C:/SpectrumIE/custom_model/Custom_Test.txt</testFilePath>

       <magicWord>DIAGNOSIS</magicWord>
       <language>English</language>

      <trainingFeatures>

 <!-- Lexical features-->
 <trainingFeature>
  <featureName>FeatureWindow</featureName>
  <featureParams>
   <entry>
    <key>Before</key>
    <value>1</value>
   </entry>
   <entry>
    <key>After</key>
    <value>2</value>
   </entry>
  </featureParams>
 </trainingFeature>

 <!-- Orthographic features-->
 <trainingFeature>
  <featureName>CaseIdentifier</featureName>
      </trainingFeature>

 <trainingFeature>
  <featureName>NumericIdentifier</featureName>
 </trainingFeature>
 </trainingFeatures>
 </trainingOptions>
```

Trainieren des Modells mit benutzerdefinieren Entitäten

Nach der Erstellung einer Optionsdatei müssen Sie mit Ihrem Modell das Identifizieren benutzerdefinierter Entitäten trainieren. Spectrum™ Technology Platform führt dies mithilfe des CLI-Befehls **iemodel train** aus. Ein trainiertes Modell wird zum Abrufen benutzerdefinierter Entitäten verwendet. Weitere Informationen zu CLI-Befehlen finden Sie im Abschnitt **Administrations-Dienstprogramm** im **Administrationshandbuch**.

Auswerten des Modells mit benutzerdefinieren Entitäten

Eventuell möchten Sie Ihr Modell nach dem Training testen, um sicherzustellen, dass die Datei mit Trainingsoptionen korrekt ist und die Entitäten wie erwartet extrahiert werden. Verwenden Sie zum Testen Ihres Modells den CLI-Befehl **iemodel trainAndevaluate model**. Weitere Informationen zu CLI-Befehlen finden Sie im Abschnitt **Administrations-Dienstprogramm** im **Administrationshandbuch**.

Extrahieren benutzerdefinierter Entitäten

Die trainierte benutzerdefinierte Entität ist nun in der Entitätenliste des **Entity Extractor**-Schrittes verfügbar und kann verwendet werden, um aus Ihren unstrukturierten Daten relevante Informationen zu extrahieren.

Informationen zu den Schritten zum Extrahieren vorher vorhandener Entitäten finden Sie unter **Extrahieren vorher vorhandener Entitäten** auf Seite 8.

3 - Beziehungsextraktion

In this section

Relationship Extractor

Die Extraktion von Beziehungen ist die Analyse des unstrukturierten Texts, um die Beziehung zwischen den verschiedenen extrahierten Entitäten zu identifizieren.

Folgende Entitätstypen werden für die Beziehungsextraktion unterstützt:

- Person
- Organisation
- Position

Die unterstützten Beziehungstypen sind:

- AffiliatedWith
- LivesIn
- OrgBasedIn
- LocatedIn
- Negativ

Beziehungstypen

RelationshipType	Typ von Entity1	Typ von Entity2	Abgedeckte Beziehungen
AffiliatedWith	*Person*	*Organization*	Zeigt eine professionelle oder akademische Beziehung zwischen den Entitäten *Person* und *Organization* an. Die Beziehung kann eine der folgenden oder eine ähnliche sein: • *Person* studiert oder hat studiert bei *Organization* • *Person* arbeitet oder hat gearbeitet mit *Organization* • *Person* wurde ein Arbeitsplatz bei *Organization* angeboten **Anmerkung:** Es folgt eine Beispielliste mit Beziehungen, die von diesem Typ abgedeckt werden. Beispiel: `James has studied from the University of Toronto and works at ABC Corp.` Hier können zwei Beziehungen geparst werden: Entity1 = `James`, RelationshipType = `AffiliatedWith`, Entity2 = `University of Toronto` Entity1 = `James`, RelationshipType = `AffiliatedWith`, Entity2 = `ABC Corp`

RelationshipType	Typ von Entity1	Typ von Entity2	Abgedeckte Beziehungen
LivesIn	*Person*	*Location*	Zeigt eine Beziehung zwischen den Entitäten *Person* und *Location* an. Die Beziehung kann eine der folgenden sein: • *Person* lebt oder lebte in *Location* • *Person* ist umgezogen nach *Location* • *Person* wurde geboren in *Location* • *Person* ist gestorben in *Location* **Anmerkung:** Es folgt eine Beispielliste mit Beziehungen, die von diesem Typ abgedeckt werden. Beispiel: `John Jamison, a National Weather Service meteorologist in Galveston, reported that a massive hurricane was about to hit the East Coast the next day.` Entity1 = `John Jamison`, RelationshipType = *LivesIn*, Entity2 = `Galveston`
OrgBasedIn	*Organization*	*Location*	Zeigt an, dass die *Organization* mindestens eines ihrer Büros am *Location* hat. Bei dem *Location* kann es sich um eine Zweigstelle, ein Entwicklungsbüro oder die Niederlassung handeln. Beispiel: `HSBC Holdings Plc. is headquartered in London, United Kingdom.` Hier können zwei Beziehungen geparst werden: Entity1 = `HSBC Holdings Plc.`, RelationshipType = *OrgBasedIn*, Entity2 = `London` Entity1 = `HSBC Holdings Plc.`, RelationshipType = *OrgBasedIn*, Entity2 = `United States of America`

RelationshipType	Typ von Entity1	Typ von Entity2	Abgedeckte Beziehungen
LocatedIn	*Location*	*Location*	Zeigt die Beziehung zwischen zwei unterschiedlichen Orten an, bei denen eine der Entitäten geografisch in der anderen enthalten ist. **Beispiel 1** `Canberra is the capital of Australia.` Hier gilt: Entity1 = `Canberra`, RelationshipType = *LocatedIn*, Entity2 = `Australia` **Beispiel 2** `India has as its capital New Delhi.` Hier gilt: Entity1 = `India`, RelationshipType = *LocatedIn*, Entity2 = `New Delhi`
Negative	*Person* *Organization* *Location*	*Organization* *Location*	Gibt an, dass keiner der oben stehenden Beziehungstypen zwischen den zwei entsprechenden Entitäten geparst werden konnte. Beispiel: `New Delhi and New York are good places to live in.` Beim Parsen dieses Eingabetexts wird keiner der unterstützten Beziehungstypen zwischen einem beliebigen identifizierten Entitätenpaar geparst. Dadurch ist eine Aufschlüsselung der Beziehungstypen vom Typ *Negative* zwischen den identifizierten Entitäten möglich: Entity1 = `New Delhi`, RelationshipType = *Negative*, Entity2 = `New York`

Anmerkung: Sie können einen **Splitter**-Schritt an die Ausgabe des **Relationship Extractor**-Schrittes anschließen, um die identifizierten Beziehungstypen und die entsprechenden Entitätenpaare, die von der Beziehung zusammengeführt wurden, zu extrahieren. Im „Splitter“-Schritt wird die hierarchische Ausgabe dieses Schrittes in eine flache Ausgabe konvertiert.

Beispiel

Wenn ein komplexer Eingabetext vorliegt, können mehrere mögliche Kombinationen von Beziehungstypen für denselben Satz gefunden werden.

Beispiel:

```
James McCarthy has settled in New York, United States as director of ABC
Technologies.
```

Wenn der **Relationship Extractor**-Schritt diesen Eingabetext unter Verwendung der in den Schrittoptionen ausgewählten Optionen parst, werden folgende Beziehungen gefunden:

Beziehung 1	Entity1 = `James McCarthy`, Entity1 Type = *Person*, RelationshipType = *LivesIn*, Entity2 = `New York`, Entity2 Type = *Location*
Beziehung 2	Entity1 = `James McCarthy`, Entity1 Type = *Person*, RelationshipType = *AffiliatedWith*, Entity2 = `ABC Technologies`, Entity2 Type = *Organization*
Beziehung 3	Entity1 = `ABC Technologies`, Entity1 Type = *Organization*, RelationshipType = *OrgBasedIn*, Entity2 = `United States`, Entity2 Type = *Location*
Beziehung 4	Entity1 = `ABC Technologies`, Entity1 Type = *Organization*, RelationshipType = *OrgBasedIn*, Entity2 = `New York`, Entity2 Type = *Location*
Beziehung 5	Entity1 = `James McCarthy`, Entity1 Type = *Person*, RelationshipType = *LivesIn*, Entity2 = `United States`, Entity2 Type = *Location*
Beziehung 6	Entity1 = `New York`, Entity1 Type = *Location*, RelationshipType = *LocatedIn*, Entity2 = `United States`, Entity2 Type = *Location*

4 - Textkategorisierung

In this section

Text Categorizer

Die Textkategorisierung, auch als Textklassifizierung bezeichnet, beinhaltet die Zuweisung von benutzerdefinierten Kategorien zu unstrukturiertem Inhalt oder Klartext (wie in E-Mails, Nachrichtenartikeln und Kommentaren), je nachdem, wie viel des Inhalts mit der Kategorie übereinstimmt. Die Kategorisierung kann auf Basis des Betreffs, Autors, Datums oder nahezu jedes definierten Klassifizierungssystems ausgeführt werden.

Sie können Ihren eigenen Kategorisierer erstellen, indem Sie ein Kategorisierungsmodell mit Ihren Daten und Kategorien trainieren. Der Trainer analysiert die Daten und speichert die erfassten Informationen im Trainingsvorgang. Er analysiert anschließend die Inhalte und bestimmt die Kategorie, zu welcher der Inhalt gehört.

Die Textkategorisierungsfunktion nutzt statistische Textkategorisierungsvorgänge. Sie wendet Machine-Learning-Methoden an, um automatische Klassifizierungsregeln zu erlernen, die auf von Menschen markierten Trainingsdokumenten basieren.

Da Sie in der Lage sind, die Kategorisierung Ihrer Wahl anzuwenden, müssen Sie zunächst Ihre Modell „trainieren“, damit es die Kategorien „erlernt“. Danach können Sie dieses Modell im **Text Categorizer**-Schritt zur Kategorisierung Ihrer unstrukturierten Daten verwenden.

Spectrum™ Technology Platform nutzt Administrationsumgebung-Befehle zur Verwaltung von Textkategorisierungsmodellen. Eine Beschreibung dieser Befehle finden Sie im Abschnitt **Administrationsumgebung** im **Administrationshandbuch**.

Vorbereiten der Daten

Der erste Schritt zur Verwendung der Textkategorisierung ist die Vorbereitung Ihrer Eingabedatei und Ihrer Testdatei. Hierzu müssen Sie die Daten in beiden Dateien als durch Tabstopp getrennte Werte gliedern. Die Dateien müssen Details im folgenden Format aufweisen:

- UFT-8-Codierung
- Durch Tabstopp getrennte Daten in zwei Spalten, wobei die erste Spalte den Kategorienamen (z. B. „Patient“ oder „Anbieter“) und die zweite Spalte die Daten für die einzelnen Kategorien (wie im nachfolgenden Beispiel dargestellt) enthält

Ihre Daten sollten wie folgt aussehen:

```
Patient     John Smith dob04181963 224 Main St. Atl GA 30311
Provider    Mark Johnson M.D. NPI5489512047 412 Washington Atl GA 30301
```

Konfigurieren von Optionen

Dies beinhaltet die Erstellung einer Datei mit `Trainingsoptionen`, die Informationen zu Ihrem Modell und die beim Training des Modells anzuwendenden Optionen enthält. Diese Datei muss im XML-Format mit UFT-8-Codierung vorliegen. Sie muss folgende Features für Header und das erforderliche Training enthalten:

Header in der Datei mit Trainingsoptionen

Der Header enthält Details zu dem Modell, dem zugehörigen Typ und dem Pfad der Test- und Eingabedateien.

- `modelName`: Der Name des Modells
- `modelType`: Der Typ des Modells (er lautet in diesem Fall `TK`, d. h. Textkategorisierung)
- `modelDescription`: Die Beschreibung des Modells
- `inputFilePath`: Der Speicherort der Eingabedatei, die für das Trainieren des Modells verwendet wird
- `testFilePath`: Der Speicherort der Testdatei

  Anmerkung:

  Die Testdatei misst die Effektivität eines Modells. Sie bestimmt das Verhalten des benutzerdefinierten Modells im Hinblick auf verschiedene Trainingsparameter. Als bewährte Methode gilt, bei dem Training oder der Auswertung Ihres benutzerdefinierten Modells unterschiedliche Eingabe- und Testdateien zu verwenden.

`algorithm`: Der Machine Learning-Algorithmus zum Trainieren des Modells (Standard ist `MaxEnt`)

Trainingsfeatures

Dies sind die Trainingsfeatures, mit denen Sie eine neue Kategorie erstellen können.

Anmerkung: Wenn Sie mehrere Features verwenden, können Sie sie in beliebiger Reihenfolge in der Datei platzieren.

- **Sprachliches Feature**: Für die Angabe der Spracheigenschaften
  - `Stemming`: Reduziert Wörter auf Ihren Wortstamm. Die Wörter „insurer“, „insured“ und „insures“ beispielsweise können alle auf den Wortstamm „insure“ reduziert werden.

```
<trainingFeature>
  <featureName>Stemming</featureName>
</trainingFeature>
```

- **Schlüsselwortfeatures**: Zum Definieren der Liste mit Schlüsselwörtern

- `IgnoreWords`: Dieses auch unter „Stoppwörter“ bekannte Feature filtert allgemeine Wörter heraus, die keine Auswirkungen auf die Kategorisierung haben, wie z. B. „der/die/das“, „und“, „aber“ usw. Diese Wörter sollten nur durch ein Komma voneinander getrennt werden, nicht durch Leerzeichen. Sie können mit diesem Feature auch den Schlüssel `Anfügen` verwenden, der zur vorhandenen Liste der Stoppwörter hinzugefügt wird, wenn er auf „true“ festgelegt ist.

```
<trainingFeature>
  <featureName>IgnoreWords</featureName>
  <featureParams>
   <entry>
    <key>WordList</key>
    <value>
     and,the,for,with,still,tri,rep,cust,keep,get,req,call
    </value>
   </entry>
   <entry>
    <key>Append</key>
    <value>True</value>
   </entry>
  </featureParams>
 </trainingFeature>
```

- `CategoryKeywords`: Gibt eine Kategorie für eine Liste mit Schlüsselwörtern an, die zu mehreren benutzerdefinierten Listen gehören. Beispiel: „Wochentage“ in der Liste `CategoryKeywords` enthält die `Schlüsselwörter` Montag, Dienstag, Mittwoch, Donnerstag und Freitag.

  Dieses Feature kann optional angeben, ob beim Abgleich die Groß-/Kleinschreibung beachtet werden soll. Bei einer Verwendung lautet der Standard `true`.

```
<trainingFeature>
 <featureName>CategoryKeywords</featureName>
 <featureParams>
  <entry>
   <key>Weekdays</key>
                    <!-- List of weekdays -->
   <value>Monday,Tuesday,Wednesday,Thursday,Friday</value>
  </entry>
  <entry>
   <key>WeekendDays</key>
                    <!-- List of weekend days -->
   <value>Saturday,Sunday</value>
  </entry>
  <entry>
   <key>CaseSensitive</key>
                    <value>True</value>
  </entry>
 </featureParams>
</trainingFeature>
```

- `KeyWords`: Sucht nach Wörtern, die Sie als zu einer benutzerdefinierten Liste gehörig angegeben haben, z. B. *DaysOfWeek* oder *Month*. Gibt zudem optional an, ob beim Abgleich die Groß-/Kleinschreibung beachtet werden soll. Bei einer Verwendung lautet der Standard „`true`“.

```
<trainingFeature>
 <featureName>KeyWords</featureName>
 <featureParams>
  <entry>
   <key>KeyWordList</key>
   <value>Monday,Tuesday</value>
  </entry>
  <entry>
   <key>CaseSensitive</key>
   <value>False</value>
  </entry>
 </featureParams>
</trainingFeature>
```

- **Lexikalisches Feature**: Für die Angabe der Eigenschaften von Lexemen
  - `NGram`: Sucht nach einem längeren Abschnitt einer Zeichenfolge, wobei „n“ für die Anzahl der zu suchenden Wörter steht. Wenn Sie beispielsweise nach dem Ausdruck „to be or not to be“ („Sein oder Nichtsein“) suchen, können Sie nach dem Unigramm „to“ oder „be“, dem Bigramm „to be“ oder „or not“, dem Trigramm „to be or“ oder „not to be“ usw. suchen.

```
<trainingFeature>
  <featureName>NGram</featureName>
  <featureParams>
   <entry>
    <key>Count</key>
    <value>3</value>
   </entry>
  </featureParams>
</trainingFeature>
```

Im Folgenden finden Sie eine Beispieldatei mit Trainingsoptionen:

```
<trainingOptions>
 <modelName>modelone</modelName>
 <modelType>TC</modelType>
 <modelDescription>modelOne</modelDescription>

<inputFilePath>C:/SpectrumIE/textclassification/train_Input.csv</inputFilePath>


<testFilePath>C:/SpectrumIE/textclassification/train_Test.txt</testFilePath>

    <algorithm>SVM</algorithm>

 <trainingFeatures>
```

```
<!-- Keyword features -->
<trainingFeature>
 <featureName>IgnoreWords</featureName>
 <featureParams>
  <entry>
   <key>WordList</key>
   <value>
    and,the,for,with,still,tri,rep,cust,keep,get,req,call
   </value>
  </entry>
  <entry>
   <key>Append</key>
   <value>True</value>
  </entry>
 </featureParams>
</trainingFeature>

<trainingFeature>
 <featureName>CategoryKeywords</featureName>
 <featureParams>
  <entry>
   <key>Category1/key>
   <value>CategoryKeyword1,CategoryKeyword2</value>
  </entry>
  <entry>
   <key>Category2/key>
   <value>CategoryKeyword3,CategoryKeyword4</value>
  </entry>
            </featureParams>
</trainingFeature>

<trainingFeature>
           <featureName>KeyWords</featureName>
      <featureParams>
  <entry>
   <key>KeyWordList</key>
   <value>
    jam,misfeed,install,help,mechanical,failure,jam,pc,connection
   </value>
  </entry>
 </featureParams>
</trainingFeature>

<!-- Linguistic feature -->
<trainingFeature>
 <featureName>Stemming</featureName>
</trainingFeature>

<!-- Lexical feature -->
<trainingFeature>
 <featureName>NGram</featureName>
 <featureParams>
```

```
   <entry>
    <key>Count</key>
    <value>3</value>
   </entry>
  </featureParams>
</trainingFeature>

</trainingFeatures>
</trainingOptions>
```

Trainieren des Modells

Nach der Erstellung einer Optionsdatei müssen Sie mit Ihrem Modell das Erkennen potenzieller vorhergesagter Beziehungen trainieren. Wenden Sie hierzu die „Machine Learning“-Methoden an. Spectrum™ Technology Platform verwendet zum Trainieren eines Modells den CLI-Befehl **iemodeltrain**. Nach dem Trainieren des Modells können Sie es bei der Kategorisierung verwenden. Weitere Informationen zu CLI-Befehlen finden Sie im Abschnitt **Administrations-Dienstprogramm** im **Administrationshandbuch**.

Auswerten des Modells

Eventuell möchten Sie Ihr Modell nach dem Training testen, um sicherzustellen, dass die Datei mit Trainingsoptionen korrekt ist und die Kategorien wie erwartet zugewiesen werden.

Sie können das Modell mit dem CLI-Befehl **iemodel trainAndevaluate model** testen. Weitere Informationen zu CLI-Befehlen finden Sie im Abschnitt **Administrations-Dienstprogramm** im **Administrationshandbuch**.

Kategorisieren von Text

1. Erstellen Sie einen Datenfluss, der einen Quellschritt wie **Read from File** oder **Read From XML**, den Schritt **Text Categorizer** sowie einen Datenladeschritt wie **Write to File** oder **Write to XML** umfasst.
2. Verweisen Sie im Quellschritt auf Ihre Eingabedatei.

3. Wählen Sie im **Text Categorizer**-Schritt das Modell im Feld **Kategorisierername** aus. Dies ist das Modell, das Sie in der Textkategorisierungsphase trainiert haben. Informationen zum Trainieren eines Modells finden Sie unter **Trainieren des Modells** auf Seite 27.
4. Wählen Sie im Feld **Kategorieanzahl** die Anzahl der Übereinstimmungsebenen der Kategorie aus, die in der Ausgabe enthalten sein soll. Zum Beispiel die höchste Übereinstimmung oder die höchste plus die zweithöchste Übereinstimmung.

   Anmerkung: Der maximale Wert entspricht der Anzahl verschiedener, beim Trainieren des Modells angegebener Kategorien.

5. Klicken Sie auf **OK**.
6. Führen Sie den Auftrag aus.

5 - Schrittreferenz

In this section

Information Extraction-Komponenten

Das Information Extraction-Modul enthält die folgenden Schritte.

- **Read From Documents**: Liest unstrukturierte Eingabedaten aus verschiedenen Dateiformaten und extrahiert die Inhalte.
- **Entity Extractor**: Extrahiert Entitäten wie Namen und Adressen aus unstrukturierten Daten, die als Zeichenfolgen übergeben werden.
- **Text Categorizer**: Weist unstrukturierten Inhalten oder Klartext (wie in E-Mails, Nachrichtenartikeln und Kommentaren) benutzerdefinierte Kategorien zu, die darauf basieren, wie viel dieses Inhalts Material aus dieser Kategorie enthält.
- **Relationship Extractor**: Extrahiert Beziehungen zwischen Entitäten.

Read from Documents

„Read from Documents“ ist ein Quellschritt, der unstrukturierte Eingabedaten aus verschiedenen Dateiformaten liest und die Inhalte extrahiert. Mögliche Quellen sind Rechtsdokumente, Kundenfeedback, Produktbewertungen, Nachrichtenartikel, Blogs, soziale Netzwerke usw. „Read from Documents“ extrahiert auch Metadaten-Felder wie Autor und Erstellungsdatum. Sobald die Daten extrahiert wurden, können sie für verschiedene Verarbeitungstypen verwendet werden, z. B. Entitätsextraktion und Zeichenfolgenmanipulation. Die Daten können auch zum Erstellen von Suchindizes für unstrukturierte Textsuchen verwendet werden.

Anmerkung: Jedes Dokument wir als ein Datensatz für diesen Schritt betrachtet.

Eingabe

Die Eingabe für den „Read from Documents“-Schritt ist eine einzelne Datei oder ein einzelner Ordner. Dieser Schritt unterstützt die folgenden Dateitypen:

- Text
- PDF
- Microsoft Outlook
- Microsoft Word
- HTML

„Read from Documents“ führt drei Arten von Extraktionen aus:

- Dokument – Gesamtes Dokument verwenden
- Seite – Eine bestimmte Seite eines Dokuments verwenden
- Auswahl – Einen bestimmten Teil eines Dokuments verwenden
- Lesezeichen – Lesezeichen aus einem PDF-Dokument verwenden

„Read from Documents“ ist Teil des Information Extraction-Moduls.

Optionen

Registerkarte „Dateieigenschaften“

Die folgende Tabelle führt die Optionen auf, die den Informationstyp steuern, der bei „Read from Documents“ zurückgegeben wird.

Tabelle 1: „ReadfromDocuments“ Optionen

Option	Beschreibung
Servername	Gibt den Namen des verwendeten Spectrum Technology Platform-Servers an.
Datei-/Ordnername	Der Pfad und Name des Quelldokuments oder -ordners. Wenn Sie auf einen Ordner verweisen möchten, verwenden Sie ein Sternchen („*“) als Platzhalterzeichen, um alle Dateien im Ordner auszuwählen. Wenn Sie auf mehrere Dateien desselben Typs innerhalb eines Ordners verweisen möchten, verwenden Sie das Platzhalterzeichen plus die Dateierweiterung („*.pdf“).
Dateityp	Der Dateityp der Quelldatei, der automatisch bei Auswahl einer Quelle ausgewählt wird: • Text • PDF • Microsoft Outlook • Microsoft Word • HTML

Option	Beschreibung	
Extraktionstyp	**Dokumentation**	Gesamtes Dokument verwenden.
	Seite	Eine bestimmte Seite eines Dokuments verwenden.
	Auswahl	Einen bestimmten Abschnitt eines Dokuments verwenden.
	Lesezeichen	Lesezeichen aus einem PDF-Dokument verwenden.
Seitenauswahl	Nur mit dem Extraktionstyp „Seite“. Alle Seiten oder einen Bereich von Seiten auswählen.	
Ausgewählte Extraktion	Nur mit dem Extraktionstyp „Auswahl“. Gibt den Suchtyp an.	
Text angeben	Nur mit dem Extraktionstyp „Auswahl“. Gibt den zu suchenden Text an.	
Anfangstext ausschließen	Nur mit dem Extraktionstyp „Auswahl“ und der Textoption „Start“. Lässt die eingegebene Zeichenfolge in den zurückgegebenen Daten weg.	
Endtext angeben	Nur mit dem Extraktionstyp „Auswahl“. Gibt den zu suchenden Endtext an.	
Endtext ausschließen	Nur mit dem Extraktionstyp „Auswahl“. Lässt die eingegebene Zeichenfolge vom Ende der zurückgegebenen Daten weg.	
Auswahlrückgabe	Nur mit dem Extraktionstyp „Auswahl“. Gibt an, wie viele Absätze für jedes Ergebnis zurückgegeben werden sollen. Wenn Sie hier beispielsweise „2“ wählen, enthalten die zurückgegebenen Daten für jedes Ergebnis den Absatz, in dem sich das Ergebnis befindet, plus den nachfolgenden Absatz, also insgesamt zwei Absätze. Der Standardwert ist 1. Nicht gültig, wenn ein Endtext angegeben ist.	

Registerkarte „Felder“

Klicken Sie auf **Erneut &generieren**, um Eingabefelder zu definieren.

Tabelle 2: Ausgabedatenoptionen

Option	Beschreibung
Attributname	Zeigt das Attribut, das dem Eingabefeld am meisten ähnelt. Wenn beispielsweise eines Ihrer Felder Datumsinformationen enthält und Sie es „Datum“ nennen, wird diesem Feld das Attribut „Datum“ zugewiesen. Diese Spalte ist nicht bearbeitbar.
Bezeichnung	Der Name des Feldes. Diese Spalte ist bearbeitbar.
Typ	Dies ist der Datentyp des Feldes.
Einschließen	Gibt an, welche Felder in einem Suchindex enthalten sein sollen.

Ausgabe

Der Schritt „Read from Documents“ verfügt über zwei ausgehende Ports. Ein Port erfasst die Daten, die vom Schritt gelesen und auf Basis der eingegebenen Kriterien zurückgegeben wurden. Es kann sich dabei um Klartext oder Metadaten (z. B. Autor, Sprache, Erstellungsdatum) handeln. Dieser Port kann mit jedem Schritt, der eingehende Daten liest (z. B. „Write to File“ oder „Write to XML“), sowie mit Primärschritten (z. B.„Validate Address“ oder „Write to Search Index“) verbunden werden. Er kann auch mit dem „Information Extractor“-Schritt verbunden werden, wenn Sie Informationen über bestimmte Entitätstypen zurückgeben möchten, die sich im Dokument befinden. Wenn Sie den Extraktionstyp „Dokument“ auswählen, enthält die Ausgabe flache Daten; bei Auswahl des Extraktionstyps „Seite“ oder „Auswahl“ enthält die Ausgabe hierarchische Daten.

Der andere Port erfasst alle Datensätze, die der Datenfluss nicht korrekt verarbeitet hat. Dieser Port wird als Fehlerport bezeichnet, und Datensätze, die durch diesen Port in das Zielsystem gelangen, werden als falsch formatiert gewertet. Das Erfassen von falsch formatierten Datensätzen hilft Ihnen, das Problem mit diesen Datensätzen zu identifizieren. Wenn Sie einen Zielschritt an den Fehlerport anhängen, enthält die resultierende Ausgabedatei alle Felder aus den fehlerhaften Datensätzen. Sie enthält auch das Feld „Reason“, das angibt, warum ein Datensatz fehlgeschlagen ist.

Tabelle 3: Unstrukturierte Reader-Ausgabe

Feldname	Beschreibung/gültige Werte
Author	Enthält in der Regel den Namen der Person, die das Dokument erstellt oder aktualisiert hat. Diese Informationen sind Teil der Metadaten des Dokuments.
Bookmark	Enthält alle Lesezeichen aus der PDF-Eingabedatei. Nur für den Extraktionstyp „Lesezeichen“.
BookmarkNo	Enthält alle Lesezeichen aus der PDF-Eingabedatei. Nur für den Extraktionstyp „Lesezeichen“.
ContentLength	Gibt die Länge des Dokuments an. Der Wert variiert je nach dem ausgewählten Extraktionstyp: **Document** Die Anzahl der Seiten im Dokument. **Page** „1“ steht für eine Einzelseite mit Inhalt.
Contents	Variiert je nach Extraktionstyp. Der Extraktionstyp „Dokument“ beispielsweise gibt das gesamte Dokument als flache Daten aus. Die Extraktionstypen „Seite“, „Auswahl“ und „Lesezeichen“ geben hierarchische Daten aus.
ContentType	Gibt den Typ des gelesenen Dokuments an, z. B. PDF, TXT usw.
Creator	Enthält in der Regel den Namen der Person, die das Dokument erstellt hat. Diese Informationen sind Teil der Metadaten des Dokuments.
Date	Gibt das Datum an, an dem das Dokument erstellt oder zuletzt aktualisiert wurde.
Keywords	Enthält beliebige Schlüsselwörter, die in den Metadaten des Dokuments angegeben wurden.
Language	Gibt die Sprache an, in der das Dokument erstellt wurde.
NPages	Gibt die Anzahl der Seiten im Dokument an.

Feldname	Beschreibung/gültige Werte
PageContents	Enthält die Inhalte der ausgewählten Seite(n). Nur für den Extraktionstyp „Seite“.
PageNo	Enthält die Seitenzahl für das Lesezeichen. Nur für den Extraktionstyp „Seite“.
Parent	Enthält den Pfad des Lesezeichens, ähnlich dem XPath einer XML-Datei. Nur für den Extraktionstyp „Lesezeichen“.
ResourceName	Gibt den Dateinamen des Dokuments an.
SectionContents	Enthält die Inhalte des ausgewählten Abschnitts. Nur für den Extraktionstyp „Auswahl“.
SectionNo	Gibt die Nummer des Abschnitts innerhalb dieses Dokuments an. Nur für den Extraktionstyp „Auswahl“.
Subject	Enthält das Thema des Dokuments, das in den Metadaten des Dokuments angegeben wurde.
Title	Enthält den Titel des Dokuments, der in den Metadaten des Dokuments angegeben wurde.

Entity Extractor

Entity Extractor extrahiert Entitäten wie Namen und Adressen aus Zeichenfolgen mit unstrukturierten Daten (auch Klartext oder Nur-Text).

Möglicherweise werden nicht alle Entitäten eines ausgewählten Typs zurückgegeben, da die Genauigkeit je nach Eingabetyp variiert. Da Entity Extractor natürlichsprachige Verarbeitung verwendet, werden bei einer Zeichenfolge mit einem grammatikalisch korrekten Satz aus einem Nachrichtenartikel oder einem Blog wahrscheinlich mehr Namen korrekt zurückgegeben, als wenn nur eine einfach Liste mit Namen und Daten vorliegt.

Eingabe

Entity Extractor akzeptiert unstrukturierte Zeichenfolgen mit Daten als Eingabe. Auch der **Read from Documents**-Schritt kann als Eingabe dienen, wenn Sie Entitäten aus einem unstrukturierten Dokument extrahieren möchten. Der **Read from Documents**-Schritt liest das Dokument und gibt Text basierend auf benutzerdefinierten Einstellungen zurück. Der **Entity Extractor**-Schritt extrahiert die erforderlichen Informationen aus diesem Text, basierend auf den ausgewählten Entitäten.

Tabelle 4: Eingabeformat

Feldname	Beschreibung
PlainText	Die unstrukturierte Zeichenfolge mit Daten, aus denen die Informationen extrahiert werden sollen.

Optionen

Über Entity Extractor-Optionen können Sie Entitäten auswählen, die darauf basieren, welche Informationen Sie aus der Eingabezeichenfolge extrahieren möchten. Standardmäßig können Sie Informationen mit *Person* und *Address* als Entitätstypen extrahieren. Sie können aber auch über die Funktion **Schnell hinzufügen** eine oder alle der 15 vorkonfigurierten Entitäten auswählen.

Name der Option	Beschreibung
System-Standardoptionen mit den folgenden Werten überschreiben	Aktivieren Sie das Kontrollkästchen, um die Standardentitätstypen *Address* und *Person* zu überschreiben. Wenn Sie das Kästchen aktivieren, wird die Schaltfläche **Schnell hinzufügen** aktiviert. Klicken Sie auf diese Schaltfläche, und wählen Sie die zum Extrahieren des Texts erforderlichen Entitäten aus. Die ausgewählten Entitäten werden zur Liste von **Entitätstypen** hinzugefügt.

Name der Option	Beschreibung
Entitätstyp	Gibt den aus der unstrukturierten Zeichenfolge zu extrahierenden Datentyp an. ***Address*** ***CreditCard*** ***Date*** ***Email*** ***HashTag*** ***ISBN*** ***Location*** ***Mention*** ***Organization*** ***Person*** ***Phone*** ***ProperNouns*** ***SSN*** ***WebAddress*** ***ZipCode***
Anzahl der Ausgabeentitäten	Gibt ab, ob eine Anzahl, wie oft eine bestimmte Entität in der Ausgabe vorhanden ist, zurückgegeben werden soll. **true** Gibt eine Anzahl der in der unstrukturierten Zeichenfolge gefundenen Entitäten zurück. **false** Gibt keine Anzahl der in der unstrukturierten Zeichenfolge gefundenen Entitäten zurück.

Ausgabe

Die Ausgabe von **Entity Extractor** ist eine Liste von übereinstimmenden Entitäten in der Eingabezeichenfolge. Wenn Sie beispielsweise einen Entitätstyp „Person“ ausgewählt haben, enthält die Ausgabe eine Liste von in der Eingabezeichenfolge gefundenen Personennamen. Wenn Sie andererseits einen **Entitätstyp** „Datum“ ausgewählt haben, enthält die Ausgabe eine Liste von in der Eingabezeichenfolge gefundenen Datumswerten.

Jede Entität, ob Name, Adresse oder Datum, wird nur einmal zurückgegeben, auch wenn die Entität mehrfach in der Eingabezeichenfolge enthalten ist.

Um zu sehen, wie oft eine Entität in der Eingabezeichenfolge enthalten ist, können Sie die Option **Anzahl der Ausgabeentitäten** im Fenster **Entity Extractor-Optionen** auswählen.

Feldname	Beschreibung
Text	Der aus der Zeichenfolge extrahierte Text.
Type	Der Entitätstyp des extrahierten Texts. Zur Auswahl stehen: ***Address*** ***CreditCard*** ***Date*** ***Email*** ***HashTag*** ***ISBN*** ***Location*** ***Mention*** ***Organization*** ***Person*** ***Phone*** ***ProperNouns*** ***SSN*** ***WebAddress*** ***ZipCode***
Count	Wenn die Option zur Rückgabe einer Anzahl aktiviert ist, enthält dieses Feld die Anzahl, wie oft eine bestimmte Entität in der Eingabe enthalten ist. Beispiel: Wenn Sie Entitäten vom Typ `Name` zurückgeben möchten und der Eingabetext fünf Instanzen mit dem Namen `John` enthält, ist der Name `John` nur einmal in der Ausgabe zu finden. Dabei ist der Entitätstyp `Name` und die Ausgabeanzahl „5“.

Relationship Extractor

Im **Relationship Extractor**-Schritt können Sie die Beziehungstypen zwischen den identifizierten Entitäten im Quellinhalt identifizieren.

Folgendes wird im **Relationship Extractor**-Schritt identifiziert:

1. Entity1
2. Typ von Entity1
3. Beziehungstyp
4. Entity2
5. Typ von Entity2

Wichtig: Der Schritt versucht, beim Identifizieren der Beziehungstypen zwischen zwei beliebigen Entitäten im Eingabetext eine möglichst hohe Genauigkeit zu erreichen. Beim Parsen von komplizierten Sätzen im Eingabetext können jedoch auch andere Beziehungen als die genaue Beziehung zwischen den beiden Entitäten identifiziert werden.

Eingabe

Der **Relationship Extractor**-Schritt verwendet natürlichsprachige Datenzeichenfolgen als Eingabe, und ermittelt die Entitäten sowie die zwischen jedem identifizierten Entitätspaar vorhandenen Beziehungstypen.

Verwenden Sie den **Read from Documents**-Schritt als Quellschritt, wenn der Eingabetext aus einem unstrukturierten Dokument stammt. Der **Read from Documents**-Schritt liest das Dokument und gibt Text basierend auf benutzerdefinierten Einstellungen zurück.

Der **Relationship Extractor**-Schritt identifiziert dann alle Entitäten und dem vorhandenen Beziehungstyp zwischen jedem Entitätspaar.

Tabelle 5: Eingabeformat

Feldname	Beschreibung
PlainText	Die unstrukturierte Datenzeichenfolge, von der Sie die vorhandenen Beziehungstypen zwischen jedem Entitätspaar identifizieren möchten.

Optionen

Mit den Optionen des **Relationship Extractor**-Schrittes können Sie angeben, welche Beziehungstypen im Eingabetext identifiziert werden sollen.

Die standardmäßig identifizierten Beziehungstypen sind:

1. *AffiliatedWith*

2. *LivesIn*
3. *OrgBasedIn*
4. *LocatedIn*

Name der Option	Beschreibung
System-Standardoptionen mit den folgenden Werten überschreiben	Aktivieren Sie das Kontrollkästchen, um die standardmäßig identifizierten Beziehungstypen zu überschreiben, und geben Sie an, welche Beziehungstypen identifiziert und aus dem Eingabetext extrahiert werden sollen. Wenn Sie das Kontrollkästchen aktivieren, ist die Schaltfläche **Schnell hinzufügen** aktiviert. Klicken Sie auf **Schnell hinzufügen**, um die Beziehungstypen auszuwählen, die im Text identifiziert werden sollen. Die ausgewählten Entitäten werden zur Liste **Beziehungstyp** hinzugefügt.

Ausgabe

Die Ausgabe von **Relationship Extractor** ist eine Liste der Beziehungssätze, die zwischen in der Eingabezeichenfolge gefundenen Entitätspaaren identifiziert wird.

Beispiel: Wenn Sie in den Schrittoptionen die Beziehungstypen *LivesIn* und *OrgBasedIn* für die Extraktion ausgewählt haben, enthält die Ausgabe eine Liste mit allen Sätzen von *Person LivesIn Location* und *Organization OrgBasedIn Location*, die im Eingabetext identifiziert wurden.

Jedes Entitätspaar mit seinem entsprechenden Beziehungstyp wird nur einmal aufgeführt.

Die für jeden extrahierten Satz von Entitäten und deren Beziehung extrahierten Informationen lauten:

Feldname	Beschreibung
Entity1	Die erste Entität von einen aus dem Eingabetext extrahierten Entitätspaar.
Entity1 Type	Der Entitätstyp der ersten Entität von dem aus dem Eingabetext extrahierten Entitätspaar. Der Entitätstyp ist einer der folgenden: • Person • Organisation • Position

Feldname	Beschreibung
Type	Der zwischen „Entity1“ und „Entity2“ identifizierte Beziehungstyp. Weitere Informationen über Beziehungstypen finden Sie unter **Beziehungstypen** auf Seite 17. **Anmerkung:** Es werden nur die in den Schrittoptionen zur Extraktion ausgewählten Beziehungstypen identifiziert und aufgeführt.
Entity2	Die zweite Entität von einen aus dem Eingabetext extrahierten Entitätspaar.
Entity2 Type	Der Entitätstyp der zweiten Entität von dem aus dem Eingabetext extrahierten Entitätspaar. Der Entitätstyp ist einer der folgenden: • Person • Organisation • Position

Text Categorizer

Dieser Schritt hilft Ihnen bei der Zuweisung von benutzerdefinierten Kategorien zu unstrukturiertem Inhalt oder Klartext (wie in E-Mails, Nachrichtenartikeln und Kommentaren), je nachdem, wie viel übereinstimmender Inhalt vorhanden ist. Der Schritt führt die definierten Kategorien auf, aus denen Sie diejenige auswählen können, die Sie für Ihre Kategorisierung benötigen. Sie müssen diese Kategorien allerdings erstellen, indem Sie ein Kategorisierungsmodell mit Ihren Daten trainieren. Details finden Sie unter **Text Categorizer** auf Seite 22.

Eingabe

Der Schritt verwendet unstrukturierte Zeichenfolgen von Daten als Eingabe. Auch der **Read from Documents**-Schritt kann als Eingabe dienen, wenn Sie Text aus einem unstrukturierten Dokument kategorisieren möchten. Der **Read from Documents**-Schritt liest das Dokument und gibt Text basierend auf benutzerdefinierten Einstellungen zurück. Dieser wird vom **Text Categorizer**-Schritt gelesen, um die gewünschte Ausgabe zurückzugeben.

Tabelle 6: Eingabeformat

Feldname	Beschreibung
PlainText	Die unstrukturierte Zeichenfolge mit Daten, aus denen die Informationen extrahiert werden sollen.

Optionen

Mithilfe der **Text Categorizer-Optionen** können Sie Parameter auswählen, je nachdem, wie Sie Ihre Eingabedatenzeichenfolge klassifizieren möchten. Sie können das Modell für die Kategorisierung sowie die gewünschte Anzahl der Übereinstimmungsebenen für die Ausgabe auswählen: zum Beispiel nur die höchste Übereinstimmung oder die höchste plus die zweithöchste Übereinstimmung.

Name der Option	Beschreibung
System-Standardoptionen mit den folgenden Werten überschreiben	Zum Überschreiben der Standardoption und Auswählen des Kategorisierers aus der Dropdown-Liste **Kategorisierername**.
Kategorisierername	Gibt das für die Textkategorisierung zu verwendende Modell an. Es werden alle Modelle aufgeführt, die Sie im Textkategorisierungsschritt trainiert haben. **Anmerkung:** Weitere Informationen finden Sie unter **Trainieren des Modells** auf Seite 27.
Kategorieanzahl	Die Anzahl der Übereinstimmungsebenen der Kategorie, die Sie in der Ausgabe wünschen. Wählen Sie beispielsweise „1“ aus, um nur die höchste Übereinstimmung anzuzeigen, und „2“, um die höchste plus die zweithöchste Übereinstimmung anzuzeigen. **Anmerkung:** Der maximale Wert entspricht der Anzahl verschiedener, beim Trainieren des Modells angegebener Klassen.

Ausgabe

Die Ausgabe führt die Kategorien auf, in die die Inhalte der Eingabezeichenfolge eingeordnet werden sowie den Rang dieser Kategorie. Der Rang kennzeichnet, wie hoch die Übereinstimmung zwischen

dem Eingabeinhalt und der Kategorie ist. „1“ bedeutet beispielsweise die höchste Übereinstimmung mit der Kategorie, „2“ bedeutet die höchste plus die nächsthöchste Übereinstimmung.

Feldname	Beschreibung
Category	Die vorhergesagte Kategorie für jeden Datensatz in der Eingabedatei.
Rank	Der Rang der Kategorien von der höchsten bis zur niedrigsten Punktzahl.

Notices

USPS® Urheberrechtshinweise

Datenbereitstellung und Hinweise

Diese CD-ROM enthält Daten einer urheberrechtlich geschützten Datenerfassung der Canada Post Corporation.

Das Geocode Address World Dataset enthält lizenzierte Daten des GeoNames-Projekts (**www.geonames.org**), die unter den Bedingungen der Creative Commons Attribution License ("Attribution License") bereitgestellt werden. Die Attribution License können Sie unter **http://creativecommons.org/licenses/by/3.0/legalcode** einsehen. Ihre Nutzung der GeoNames-Daten (wie im Spectrum™ Technology Platform Nutzerhandbuch beschrieben) unterliegt den Bedingungen der Attribution License. Bei Konflikten zwischen Ihrer Vereinbarung mit Pitney Bowes Software, Inc. und der Attribution License hat die Attribution License lediglich bezüglich der Nutzung von GeoNames-Daten Vorrang.

3001 Summer Street
Stamford CT 06926-0700
USA

www.pitneybowes.com