

“The Bad Part of My Chatbot Experience”

NLP技术到底哪家强?

需求背景

一句话中识别出
* 时间
* 主题
* 地点



讯飞开放平台

词法分析: <https://www.xfyun.cn/services/lexicalAnalysis>

关键字提取: <https://www.xfyun.cn/services/keyword-extraction>

The screenshot shows the homepage of the XFLY Open Platform. The top navigation bar includes links for products, solutions, document library, iFLYOS, service market, AI university, AI competition, and ecosystem platform. A search bar is located at the top right. The main content area is divided into several sections:

- 精选商家**: A banner featuring various AI-related services.
- 智能语音**: A large section with icons for speech recognition, synthesis, extension, natural language processing, and basic services.
- 文字识别**: Includes options for handwriting, printed text, and document analysis.
- 人脸识别**: Options for facial verification, comparison, watermark detection, and feature analysis.
- 内容审核**: Options for filtering色情 (adult), 政治 (politics), 暴恐 (terrorism), and 敏感 (sensitive) content.
- 语音硬件**: Options for microphones and chips.
- 图像识别**: Options for scene and object recognition.
- 基础服务**: Options for IoT, smart hardware, advertising, messaging, and push notifications.
- 医疗产品**: Options for medical audio, microphones, and diagnostic robots.

At the bottom, there is a news feed with four items:

- 18 / 07: 广告过滤 (New). Description: A.I. 超值福利, 等你来拿! (A.I. Super Value Benefit, wait for you to get it!).
- 13 / 06: 智能语音的未来, 从这里开始! (The future of intelligent voice, starts here!).
- 6 / 06: 5G牌照正式发放! 网友: 我… (5G license officially issued! Netizens: I...).
- 6 / 06: AI预测世界杯? 算了, 足球比… (AI predict World Cup? Forget it, football is better than...).

词法分析

基于大数据和用户行为，提供分词、词性标注、命名实体识别，定位基本语言元素。

[立即体验](#)

双麦克风阵列
六麦环形阵列
语音合成芯片
xTTS-C离线语音合成
离线识别模块
双麦语音交互模块

是指为自然语言文本中的每个词汇赋予一个词性的过程

- 命名实体识别 (Named Entity Recognition, NER)

是在句子的词序列中定位并识别人名、地名、机构名等实体



人脸识别

人脸验证与检索
人脸比对
人脸水印照比对
静默活体检测
人脸特征分析

{ 功能体验 }

请输入一段需要分析的文本: [换一个示例](#)

今天晚上6点晨晨英语课，在商务会馆1101

文字识别

手写文字识别
印刷文字识别
印刷文字识别(多语种)

体验版最多输入 **100**字

[开始](#)

名片识别

身份证识别
银行卡识别
营业执照识别
增值税发票识别
拍照速算识别

分析结果:

分词 & 词性标注:

今天 晚上 6点 晨晨 英语 课 , 在 商务 会馆 1101

标点 人名 名词 数词 其他专名 介词 时间词

内容审核

产品&服务

语音合成

在线语音合成

离线语音合成

语音识别

语音听写

语音转写

实时语音转写

离线语音听写

语音唤醒

离线命令词识别

语音扩展

A.I.客服平台能力中间件

语音评测

语义理解

性别年龄识别

声纹识别

歌曲识别

语音硬件

双麦克风阵列

六麦环形阵列

语音合成芯片

xTTS-C离线语音合成

离线识别模块

关键词提取

关键词提取是把文本中包含的信息进行结构化处理，并将提取的信息以统一形式集成在一起。

[立即开通](#)

活动推广期间，接入即享2万次/日免费服务

服务提供商 云孚科技



{ 功能体验 }

请输入一段需要分析的文本: [换一个示例](#)

今天晚上6点晨晨英语课，在商务会馆1101

体验版最多 100 字

[开始](#)

分析结果

名称	商务	会馆	晨晨	6点	晚上	今天	英语	1101
权重	0.684	0.675	0.664	0.563	0.548	0.525	0.500	0.500

百度开放平台

词法分析: https://ai.baidu.com/tech/nlp_basic/lexical

The screenshot shows the Baidu AI Open Platform interface. The top navigation bar includes links for Gmail, Google, g翻译, baidu, 毛豆少儿课堂, 百度词典, editer, painter, 毛豆网, live, site:ke, ke, pandao, Canon In D, github, and 百度统计. Below the bar, there's a secondary navigation menu with categories like 大脑 | AI开放平台, 开放能力, 开发平台, 行业应用, 生态合作, AI市场, 开发与教学, 资讯, 社区, and a search bar.

The main content area is divided into several sections:

- 技术能力** (Technology Capabilities):
 - 语音技术: 词法分析 **热门**
 - 图像技术: 词向量表示
 - 文字识别: 词义相似度
 - 人脸与人体识别: 依存句法分析
 - 视频技术: DNN语言模型
 - AR与VR
 - 自然语言处理 >** (highlighted):
 - 文本审核 **热门**
 - 数据智能
 - 知识图谱
 - 场景方案
 - 部署方案
- 语言处理基础技术**:
 - 文本纠错
 - 情感倾向分析 **热门**
 - 评论观点抽取
 - 对话情绪识别
 - 文本标签 **热门**
 - 文章分类
 - 新闻摘要 **新品**
- 语言处理应用技术**:
 - 通用翻译API
 - 定制化翻译API
 - 语音翻译SDK
 - 拍照翻译SDK
 - 垂直领域API
 - 语种识别API
 - AI同传 **邀测**
- 机器翻译**
- 开发平台**:
 - EasyDL定制化文本分类
 - 对话系统定制平台UNIT
 - 内容审核平台
 - 智能创作平台
- 场景方案**:
 - 机器人平台ABC Robot
- 部署方案**:
 - 私有化AI模型
- 工具集**:
 - NLP开源工具

体验区

请输入一段想分析的文本： [随机示例](#)

今天晚上6点晨晨英语课，在商务会馆1101

分析结果：
[对结果不满意？](#)

分词词性

今天晚上6点 TIME	晨晨 nr	英语 n	课 n	,	在 p	商务会馆 LOC	1101 m
----------------	----------	---------	--------	---	--------	-------------	-----------

词汇详情

词汇：商务会馆
词性：地名
实体识别：地名

专名识别

地名

时间



实际使用的库和效果

@microsoft/recognizers
BosonNLP NER
chi-time-nlp

@microsoft/recognizers

npm安装包: <https://www.npmjs.com/package/@microsoft/recognizers-text-suite>

Noble Pug Mullet

npm Search packages

Join us for the upcoming webinar, "Enterprise JavaScript done right: the tools you love, the security you need." [Sign up here »](#)

[npm Enterprise](#) [Products](#) [Solutions](#) [Resources](#) [Docs](#) [Support](#)

[Search](#) [Join](#) [Log In](#)

@microsoft/recognizers-text-suite

1.1.4 • [Public](#) • Published 9 months ago

[Readme](#) [6 Dependencies](#) [11 Dependents](#) [9 Versions](#)

Microsoft.Recognizers.Text for JavaScript

Getting Started

Recognizer's are organized into groups and designed to be used in C#, Node.js, Python and Java to help you build great applications! To use the samples, install the `recognizers-text-suite` package, or clone our GitHub repository using Git.

Cloning and building the Repository

```
git clone https://github.com/Microsoft/Recognizers-Text.git
cd Recognizers-Text
```

install

```
> npm i @microsoft/recognizers-text-s...
```

weekly downloads

4,113 

version [1.1.4](#) license [MIT](#)

open issues [88](#) pull requests [7](#)

homepage repository

recognizer 返回值

```
maodou-nlp.js results after recognizeDateTime [ { start: 0,
  end: 5,
  resolution: { values: [Array] },
  text: '今天晚上6点',
  typeName: 'datetimeV2.datetime' } ] +62ms
maodou-nlp.js {
  maodou-nlp.js   "start": 0,
  maodou-nlp.js   "end": 5,
  maodou-nlp.js   "resolution": {
    maodou-nlp.js     "values": [
      maodou-nlp.js       {
        maodou-nlp.js         "timex": "2019-07-26T18",
        maodou-nlp.js         "type": "datetime",
        maodou-nlp.js         "value": "2019-07-26 18:00:00"
      }
    ]
  },
  maodou-nlp.js   "text": "今天晚上6点",
  maodou-nlp.js   "typeName": "datetimeV2.datetime"
maodou-nlp.js } +0ms
maodou-nlp.js getTimeInResult() { results:
  [ { start: 0,
    end: 5,
    resolution: [Object],
    text: '今天晚上6点',
    typeName: 'datetimeV2.datetime' } ] } +0ms
maodou-nlp.js time: 2019-7-26 6:00:00 PM +1ms
[parser] ==> Time: { time: 2019-07-26T10:00:00.000Z }
```

```
18
19     // if results have date and time, just return value
20     result = results.find(x => x.typeName === 'datetimeV2.datetime')
21     if (result) {
22         if (result.resolution.values[1])
23             datetime = new Date(result.resolution.values[1].value)
24         else
25             datetime = new Date(result.resolution.values[0].value)
26
27         return datetime
28     }
29
30     // first deal with time
31     time = results.find(x => x.typeName === 'datetimeV2.time')
32     var timeStr
33     if (time) {
34         timeStr = time.resolution.values[0].value
35         if (time.resolution.values[1] && time.resolution.values[1] < "23:59:59") // we prefer the later time for kid's
36             timeStr = time.resolution.values[1].value
37     } else {
38         time = results.find(x => x.typeName === 'datetimeV2.timerange')
39         if (time) {
40             timeStr = time.resolution.values[0].start
41         } else {
42             // do nothing with timeStr
43         }
44     }
45     debug({timeStr})
46
47     if (!timeStr) {
48         // if results have date and time range, we return start
49         result = results.find(x => x.typeName === 'datetimeV2.datetimerange')
50         if (result) {
51             if(result.resolution.values[1])
52                 datetime = new Date(result.resolution.values[1].start)
53             else
54                 datetime = new Date(result.resolution.values[0].start)
55
56             return datetime
57         } else
58             return // undefined
59     }
```

BosonNLP

命名实体识别: <http://docs.bosonnlp.com/ner.html>

[BosonNLP HTTP API](#)

Search docs

快速上手指南

单文本分析

情感分析

命名实体识别

依存文法分析

关键词提取

新闻分类

语义联想

分词与词性标注

时间转换

新闻摘要

多文本分析

应用场景示例

其他

Docs » 命名实体识别

命名实体识别

命名实体识别（NER）是指识别文本中具有特定意义的实体，主要包括人名、地名、机构名、专有名词等。命名实体识别是信息提取、问答系统、句法分析、机器翻译等应用领域的重要基础工具，作为结构化信息提取的重要步骤。

在 BosonNLP NER 中，我们将识别以下类别的实体：

时间	time
地点	location
人名	person_name
组织名	org_name
公司名	company_name
产品名	product_name
职位	job_title

npm install bosonnlp

npm安装包: <https://www.npmjs.com/package/bosonnlp>

bosonnlp

0.1.0 • Public • Published 2 years ago

[Readme](#) [0 Dependencies](#) [0 Dependents](#) [10 Versions](#)

BosonNLP

BosonNLP is a node sdk for <http://bosonnlp.com>.

[build passing](#) [npm v0.1.0](#) [downloads 4.8k](#)

Installation

```
$ npm install bosonnlp
```

Usage

```
var bosonnlp = require('bosonnlp');
var nlp = new bosonnlp.BosonNLP('YOUR_API_KEY');
nlp.ner('成都商报记者 姚永忠', function (result) {
  console.log(result);
});
// [{"tag": ["ns", "n", "n", "nr"],
//   "word": ["成都", "商报", "记者", "姚永忠"],
//   "entity": [[0, 2, "product_name"], [3, 4, "person_name"]]}]
```

install

```
> npm i bosonnlp
```

weekly downloads

23 

version	license
0.1.0	ISC

open issues

4

pull requests

3

homepage	repository
github.com	 github

last publish

2 years ago

collaborators



Boson NER 返回值

```
maodou-nlp.js result: [{"word": ["今天", "晚上", "6点", "晨晨", "英语", "课", "", "", "在", "商务", "会馆", "1101"],  
"tag": ["t", "t", "t", "v", "nz", "n", "wd", "p", "n", "n", "m"], "entity": [[0, 3, "time"]]}] +108ms  
maodou-nlp.js result[0]: { word:  
  [ '今天', '晚上', '6点', '晨晨', '英语', '课', ' ', ' ', '在', '商务', '会馆', '1101' ],  
  tag: [ 't', 't', 't', 'v', 'nz', 'n', 'wd', 'p', 'n', 'n', 'm' ],  
  entity: [ [ 0, 3, 'time' ] ] } +0ms  
maodou-nlp.js [index, word, tag] -> 0 今天 t +1ms  
maodou-nlp.js [index, word, tag] -> 1 晚上 t +0ms  
maodou-nlp.js [index, word, tag] -> 2 6点 t +0ms  
maodou-nlp.js [index, word, tag] -> 3 晨晨 v +0ms  
maodou-nlp.js [index, word, tag] -> 4 英语 nz +0ms  
maodou-nlp.js [index, word, tag] -> 5 课 n +0ms  
maodou-nlp.js [index, word, tag] -> 6 , wd +0ms  
maodou-nlp.js [index, word, tag] -> 7 在 p +0ms  
maodou-nlp.js [index, word, tag] -> 8 商务 n +0ms  
maodou-nlp.js [index, word, tag] -> 9 会馆 n +0ms  
maodou-nlp.js [index, word, tag] -> 10 1101 m +0ms  
[parser] ==> Title: { title: '晨晨英语课' }  
[parser] ==> Location: { location: '英语课商务会馆1101' }
```

源码: <https://github.com/maodouio/wechaty-getting-started/blob/master/examples/third-party/maodou/maodou-nlp.js>

```
122 ▼      b_nlp.ner(input, function (result) {  
123          debug('result:', result);  
124  
125          var b_result = JSON.parse(result)  
126          debug('result[0]:', b_result[0]);  
127 ▼          if (b_result[0]) {  
128              const length = b_result[0]["word"].length  
129              for (var i = 0; i < length; i++)  
130                  debug('[index, word, tag] -> ', i, b_result[0]["word"][i], b_result[0]["tag"][i])  
131  
132          //debug('result[0] ["word"]', b_result[0] ["word"])  
133 ▼          title = b_result[0] ["word"].filter((x,index) =>  
134              b_result[0] ["tag"][index] === 'n' ||  
135              b_result[0] ["tag"][index] === 'nl' ||  
136              b_result[0] ["tag"][index] === 'nz' ||  
137              b_result[0] ["tag"][index] === 'v' ||  
138              b_result[0] ["tag"][index] === 'vi' ||  
139              b_result[0] ["tag"][index] === 's')  
140              .slice(0, 3)  
141              .join('')  
142  
143 ▼          location = b_result[0] ["word"].filter((x,index) =>  
144              b_result[0] ["tag"][index] === 'ns' ||  
145              b_result[0] ["tag"][index] === 'nt' ||  
146              b_result[0] ["tag"][index] === 'nz' ||  
147              b_result[0] ["tag"][index] === 'an' ||  
148              b_result[0] ["tag"][index] === 'n' ||  
149              b_result[0] ["tag"][index] === 'm' ||  
150              b_result[0] ["tag"][index] === 'q' ||  
151              b_result[0] ["tag"][index] === 's')  
152              .slice(0, 5)  
153              .join('')  
154  
155          // [2, 9, "location"]  
156          var location_array = b_result[0].entity.filter(item => item.indexOf("location")>=0)  
157  
158 ▼          if (location_array.length > 0) {  
159              // clear the old location value  
160              location = ""  
161  
162 ▼              for (var i = 0; i < location_array.length; i++) {  
163                  const from = location_array[i][0]  
164                  const to = location_array[i][1]  
165                  //debug({from}, {to})  
166  
167                  const l = b_result[0] ["word"].slice(from, to).join('')
```

Time-NLP

代码库: <https://github.com/shinyke/Time-NLP>

The screenshot shows a Google search results page with the query "NLP 时间识别". The results are as follows:

- 第六章 (1.4) 自然语言处理实战——时间语义抽取- 两只橙的博客- CSDN**
https://blog.csdn.net/lzc4869/article/details/79528931 ▾ Translate this page
Mar 12, 2018 - 本工具是由复旦 NLP 中的时间分析功能修改而来，做了一些细节和功能的优化，经 ... 同时，结束时间也继承上文时间，识别到下月1号下午5点。
- NLP实体命名识别之时间识别- qq_38923076的博客- CSDN博客**
https://blog.csdn.net/qq_38923076/article/details/81605085 ▾ Translate this page
NLP实体命名识别之时间识别. 2018年08月14日21:11:23 碧空之戈 阅读数1553. 本程序针对的是酒店的预定系统，已经将语音转换为中文文本的情况下，将时间转换 ...
- shinyke/Time-NLP: 中文语句中的时间语义识别。即通过分析 ... - GitHub**
https://github.com/shinyke/Time-NLP ▾ Translate this page
中文语句中的时间语义识别。即通过分析中文语句，识别出话语中提到的时间。. Contribute to shinyke/Time-NLP development by creating an account on GitHub.
- Python自然语言处理实战：日期实体提取- 知乎**
https://zhuanlan.zhihu.com/p/39088702 ▾ Translate this page
Jul 8, 2018 - 今天我要介绍的案例是自然语言处理中最为常见的：关键实体识别- ... 这里面就用到Jieba词性标注的功能，提取其中“m”（数字）“t”（时间）词性的词。
- 一知大脑|一知智能NER时间实体识别取得行业性突破- 知乎**
https://zhuanlan.zhihu.com/p/70695451 ▾ Translate this page
Jun 24, 2019 - 近期，一知智能研究院在主攻的自然语言处理（NLP）领域取得了“行业级”的突破。针对市面上智能客服普遍存在的“时间识别障碍”，我们给出了基于“ ...

chi-time-nlp

npm安装包：<https://github.com/JohnnieFucker/ChiTimeNLP>

The screenshot shows the npm search interface with the query 'Time NLP' entered in the search bar. The results page displays 8 packages found, sorted by Maintenance. The packages listed are:

- @microsoft/recognizers-text-date-time**
Recognizers-text provides robust recognition and resolution of date/time expressed in multiple languages.
Published by  **recognizers.text** • 1.1.4 • 9 months ago
Tags: nlp, nlp-entity-extraction, datetime, entity-extraction, timex, numex, parser-library
- chi-time-nlp**
中文语句中的时间语义识别。即通过分析中文语句，识别出话语中提到的时间。参考<https://github.com/shinyke/Time-NLP>
Published by  **johnniefucker** • 1.0.4 • 2 years ago
Tags: time, nlp
- @microsoft/recognizers-text-suite**
Recognizers-text-suite provides robust recognition and resolution of numbers, units, date/time, and more; expressed in multiple languages.
Published by  **recognizers.text** • 1.0.4 • 2 years ago
Tags: nlp, nlp-entity-extraction, datetime, entity-extraction, timex, numex, parser-library

At the bottom of the page, there is a cookie consent banner with the following text:
This website stores cookies on your computer. These cookies are used to collect information about how you interact with our website and allow us to remember you. We use this information in order to improve and customize your browsing experience and for analytics and metrics about our visitors both on this website and other media. To find out more about the cookies we use, see our Privacy Policy.

If you decline, your information won't be tracked when you visit this website. A single cookie will be used in your browser to remember your preference not to be tracked.

Buttons: Accept (dark blue), Decline (light blue)

chi-time-nlp

源码: <https://github.com/maodouio/wechaty-getting-started/blob/master/examples/third-party/maodou/maodou-nlp.js>

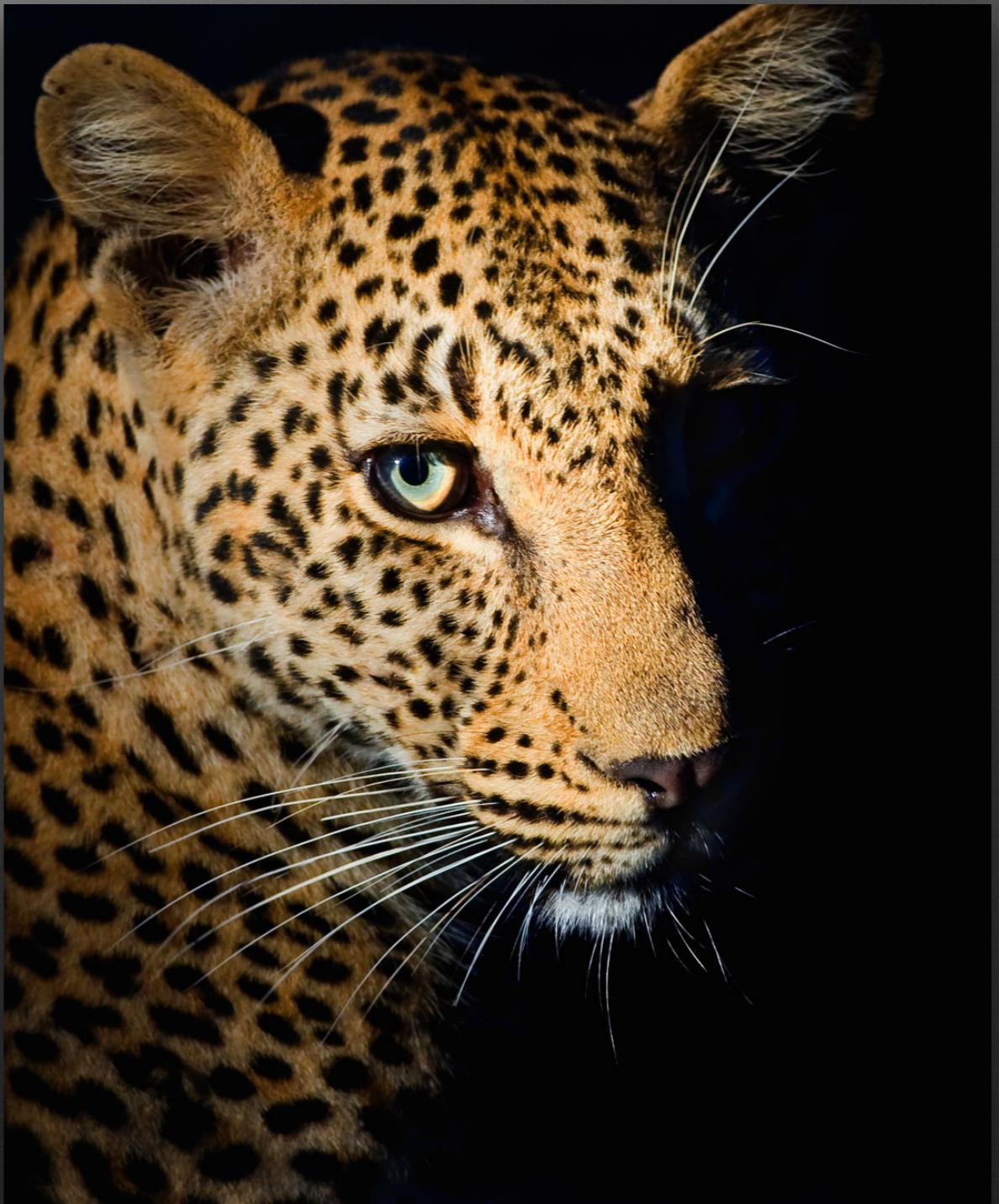
```
5
6  // const NLP = require('chi-time-nlp')
7  // var nlp = new NLP()
8
```

```
90 ▼ function parseTime(input) {
91     debug('input before recognizeDateTime', input)
92
93     // no longer using ChiTimeNlp
94     // const time = nlp.parse(input)
95     // return time
96
```

```
3  var NLP = require('./index');
4
5  var test = ['没有时间点', 'Hi, all. 下周一下午三点开会', '周一开会', '周五开会', '下下周一开始',
6  var n = new NLP();
7  test.forEach(function (s) {
8      console.log(s, '====>', n.parse(s));
9  });
10
```

理想中的API是这样的

- 搜索排在首页
- 接口和返回值简单清晰
- 无需注册Key/Token
- 自带测试调用范例代码
- 部署一个Demo网页



“有一天微信SDK开放了 `wechat.say()` 怎么办”

– *Wechaty*面临的挑战和机会

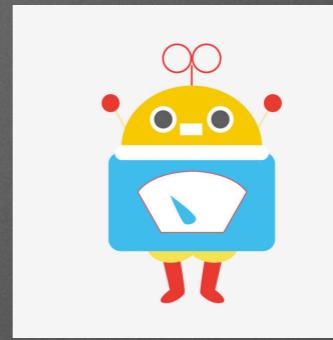
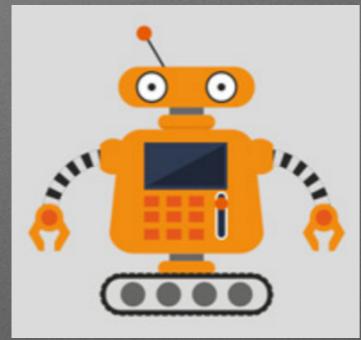
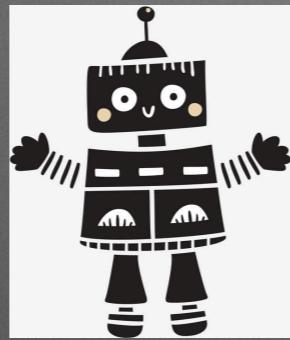
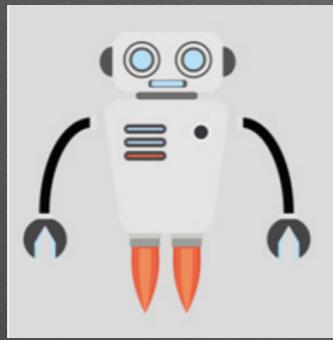
“像统一小程序开发一样统一chatbot”

- *BotHub* 之畅想 *Chateleon vs Chameleon*(变色龙)

“给Chatbot开发做一套最好的智能AI接口吧”

– *Chaty.ai vs NLP技术到底哪家强*

GitHub - Git BotHub - Chaty.ai



BotHub
(代码托管, 开发社区)

Chaty.ai
(统一的智能AI接口)

Chateleon

Wechaty Open.Feishu DD.Chat ...
(微信平台) (飞书平台) (钉钉平台) ...

- Future of Chatbot Friday