STeP-1: Standard Text preparation for Persian language

راهنمای استفاده از توابع مربوط به ابزار STeP-1

در این سامانه دو تابع زیر برای استفاده از این ابزار ایجاد شده است:

- Tokenize (قطعه بند)
- Stem (ریشه یاب)

خروجی این توابع به فرمت جیسون (json) بوده و دارای ساختار زیر است :

```
{
   string message;
   bool result;
   object Data;
}
```

Message حاوی پیام سیستم می باشد. در صورتی که تابع با موفقیت به اجرا برسد، پیام زیر را نمایش می دهد: می‌گیرد "باموفقیت انجام شد" و مقدار Result برابر true خواهد بود و مقدار فیلد دیتا شامل اطلاعات مربوط به نتیجه اجرا تابع می‌باشد. و اگر اجرای تابع موفق نباشد مقدار فیلد Message شامل پیام خطا و فیلد result برابر false خواهد بود.

راهنمای استفاده از برنامه قطعه بند

سیستم قطعه بند، متن را به کلمات و جملات تشکیل دهنده‌اش تجزیه می‌کند. در این سیستم فاصله‌ها و نیم فاصله‌ها بین کلمات فارسی تصحیح می‌شود. همچنین این سیستم، متن را تا حدی بر اساس اصول نگارشی فرهنگستان زبان و ادب فارسی ویرایش می‌کند. در جدول زیر نمونه هایی از ورودی و خروجی برنامه آورده شده است.

ورودی	خروجی
اورادر مدرسه اش خوش حال دیدم !	[او را در، اورادر] مدرسه‌اش خوشحال دیدم!
در24/5سالگی شرکتIGFراتأسیس کرد .	در 24/5 سالگی شرکت IGF [را تأسیس، راتأسیس] کرد.
کتابهاتر شده اند.	[کتاب‌ها تر، کتابهاتر] شده‌اند.
جام می بیافتاد .	جام می بیافتاد.
ازتومیپرسند به کجا می روی ؟	[از تو می‌پرسند، ازتومیپرسند] به کجا می‌روی؟

ساعت 2 :30دقیقه ،به صرافی جی .اف. سی رفتم و45, 000تومان بابت دفترچه یشان پرداختم .	ساعت 2:30 دقیقه، به صرافی جی.اف.سی رفتم و 45,000 تومان بابت دفترچه‌یشان پرداختم.

این تابع متن ورودی را گرفته (inputText) و متن اصلاح شده ،کلمات ، تمام جملات متن و کلمات برچسب خورده را به عنوان خروجی برنامه برگردانده می شود.

تابع دارای ساختار زیر می باشد.

آدرس پایه	http://nlp.sbu.ac.ir:40101/Api/Client
نام تابع	Tokenize
نوع فراخوانی	Post

تابع دارای پارامترهای ورودی زیر است:

نام پارامتر	نوع پارامتر	شرح
inputString	String	متن ورودی
dontCorrectInsideQuote	Boolean	اگر این پارامتر برابر true قرار داده شود، متن داخل double quote اصلاح نمی‌شود.
correctDelimOnly	Boolean	اگر این پارامتر برابر true قرار داده شود، تنها علائم نگارشی در متن صحیح می‌شوند.
goToSegmentor	Boolean	اگر این پارامتر برابر true قرار داده شود، در مواردی که کلمه تشخیص داده نمی‌شود، کلمه وارد تابع segmentor می‌شود. این تابع کلمه را به اجزای با معنی تجزیه می‌کند. مانند مثال 1 در جدول بالا که "اورادر" را در خروجی به صورت [او را در، اورادر] نوشته است.
firstCaseOnly	Boolean	اگر این پارامتر برابر true قرار داده شود، در موارد مبهم تنها اولین حالت پیدا شده توسط برنامه، نشان داده می‌شود. مثلا در مورد "اورادر" فقط "او را در" در خروجی نمایش داده می‌شود.
correctCompoundWords	Boolean	اگر این پارامتر برابر true قرار داده شود، کلمات مرکب تشخیص داده شده توسط برنامه، در خروجی اصلاح می‌شوند

به صورت پیش فرض بهتر است این تابع با مقادیر زیر اجرا شود :

```
"InputString": "inputString"
"dontCorrectInsideQoute": false,
"correctDelimOnly" :false,
"goToSegmentor" : false,
"firstCaseOnly" : true,
"correctCompoundWords":false,
```

صدا زدن تابع به صورت زیر باعث درست شدن برخی از کلمات مرکب و اشکالات تایپی که در اثر اتصال چند کلمه به هم در متن وجود دارد (مانند "اورادر") میشود. البته در این صورت سرعت اجرای برنامه نیز کندتر می‌شود.

```
"InputString": "inputString"
"dontCorrectInsideQoute": false,
"correctDelimOnly" :false,
"goToSegmentor" : true,
"firstCaseOnly" : true,
"correctCompoundWords":true,
```

همچنین برای استفاده بهتر از سامانه بهتر است طول متن ورودی در هر فراخوانی از صدهزار کاراکتر بیشتر نباشد. چنان که اشاره شد، درصورت فراخوانی در حالت تصحیح کلمات مرکب، این محدودیت نیز با توجه به متن ورودی ممکن است بیشتر شود.

ساختار خروجی کلمات برچسب خورده Morph است، که شامل موارد زیر می باشد:
در خروجی، اولین Morph پیدا شده برای هر کلمه از متن در یک لیست برگردانده می شود. هر کدام از عناصر این لیست از نوع Morph می باشند.

```
public struct Morph
{
    public string word;
    public List<string> prefixs;
    public List<string> postfixs;
    public string stem;
    public string tag;
    public string kind;
    public string tense;
    public int frequency;
    public string phonetic;
};
```

- نحوه فراخوانی:

http://nlp.sbu.ac.ir:40101/Api/Client/Tokenize

- نمونه ورودی:

```
{
            "InputString":"سلام",
            "dontCorrectInsideQoute": false,
            "correctDelimOnly" :false,
            "goToSegmentor" : false,
            "firstCaseOnly" : true,
            "correctCompoundWords":true,
        }
```

- نمونه خروجی:


```
{
    "message": "باموفقیت انجام شد",
    "result": true,
    "data": {
        "inputString": "سلام",
        "tokenOutput": "سلام",
        "taggedSubstrings": [
            {
                "kind": "اسم",
                "prefixs": [],
                "postfixs": [],
                "word": "سلام",
                "frequency": 270,
                "phonetic": "salAm",
                "num": 0,
                "stem": "سلام",
                "tag": "N1",
                "tense": "",
                "zamirMafoliChasbide": ""
            }
        ],
        "sentences": [
            "سلام"
        ],
        "words": [
            "سلام"
        ]
    }
}
```


راهنمای استفاده از برنامه ریشه یاب

سیستم ریشه یاب قادر به ریشه یابی تمام کلمات تصریفی، تعدادی از کلمات اشتقاقی و تحلیل ساختواژی آن‌هاست. این سیستم به زبان C# نوشته شده است. در جدول زیر نمونه هایی از کلماتی که این سیستم قادر به ریشه یابی است، آورده شده است.

آینده منفی	نخواهم خواند
ساده	خواندم

ساده منفی	نخواندم
ماضی استمراری	می خواندم
ماضی استمراری منفی	نمی خواندم
ماضی استمراری با داشت	داشتم می خواندم
ماضی نقلی	خوانده ام

علامت جمع "ان"	زنان
علامت جمع "گان"	پرندگان
علامت جمع "یان"	دانایان
علامت جمع "ات"	محصولات
علامت جمع "جات"	ترشیجات
علامت جمع "ون"	روحانیون
علامت جمع "ین"	معلمین

انه	ماهرانه
گی	زندگی
آگین	زهرآگین
ا	ژرفا
ه	کناره
ستان	گلستان
ین	سنگین
ینه	نقدینه
چی	درشکه چی
چه	ناوچه

این سیستم قادر به ریشه یابی افعال مرکب مانند " آماده شدیم" و کلمات مرکب مانند "کلاس بندی" نمی باشد.

تابع ریشه یاب کلمه ورودی را گرفته و لیستی از مورفولوژی های پیدا شده برای کلمه را برمی‌گرداند. هر کدام از عناصر این لیست از نوع Morph می باشند. Morph ساختاری است که شامل موارد زیر می باشد:

```
public struct Morph
{
    public string word;
    public List<string> prefixs;
    public List<string> postfixs;
    public string stem;
    public string tag;
    public string kind;
    public string tense;
    public int frequency;
```

```
        public string phonetic;
    };
```

تابع ریشه یاب، تمام ریشه‌های تصریفی و اشتقاقی پیدا شده برای کلمه را برمی‌گرداند.

تابع دارای ساختار زیر می باشد.

آدرس پایه	http://nlp.sbu.ac.ir:40101/Api/Client
نام تابع	Stem
نوع فراخوانی	Post

تابع دارای پارامترهای ورودی زیر است:

نام پارامتر	نوع پارامتر	شرح
inputString	String	کلمه ورودی

- نحوه فراخوانی :

http://nlp.sbu.ac.ir:40101/Api/Client/Stem

- نمونه ورودی:

```
{
    "InputString":"سلام",
}
```

- نمونه خروجی:


```
{
    "message": "باموفقیت انجام شد",
    "result": true,
    "data": {
        "output": "سلام",
        "stemmer_Morph": "اسم :نوع\r\nصورت واجی ریشه: salAm\r\n270 :بسامد ریشه\r\nسلام\r\n\r\n\r\n**********\r\n\r\n\r\n",
        "morphs": [
            {
                "word": "سلام",
                "prefixs": [],
                "postfixs": [],
                "stem": "سلام",
                "tag": "N1",
                "kind": "اسم",
                "tense": "",
                "frequency": 270,
                "phonetic": "salAm",
                "hasZamir_mafooli_chasbide": false,
                "zamir_mafooli_chasbide": "",
                "num": 0
            }
        ]
    }
}
```