Treebanks, Linguistic Theories and Applications

Multilingual Treebanks: the Universal Dependencies Case

Lecture Three

Petya Osenova and Kiril Simov
Sofia University “St. Kliment Ohridski”, Bulgaria
Bulgarian Academy of Sciences, Bulgaria
ESSLLI 2018
Plan of the Lecture

• Constituent-to-Dependency Conversion
• Universal Dependencies (UD): Overview
• UD: main principles
• UD: treebanks
• BulTreeBank in UD
• UD: potential
• Universal Dependencies link: http://universaldependencies.org/

Some slides are taken from the presentation of Nathan Schneider, Amir Zeldes “Corpus Linguistics Treebanks and Dependencies”, Linguistic Institute 2017, Georgetown University
Constituent-to-Dependency Conversion

![Diagram of Constituent and Dependency Structures]

The diagram illustrates the transformation from a constituent tree to a dependency tree, highlighting the systematic conversion of syntactic constituents into dependency relations.
Constituent-to-Dependency Conversion Method

The procedure for conversion includes the following steps:

• Assign each constituent in the parse tree a unique head selected amongst the constituent’s children
  
  \[
  \begin{align*}
  \text{NP} & \rightarrow \text{DT NN*} \\
  \text{VP} & \rightarrow \text{VBD* NP} \\
  \text{S} & \rightarrow \text{NP VP*}
  \end{align*}
  \]

• Creation of a *head percolation table*, a set of priority lists, to find heads of constituents

• Define a set of heuristics to infer arc labels in the dependency tree (Implemented in PENN2MALT)
Constituent-to-Dependency Conversion Method (Example)

Figure 1: A constituent tree from the Penn Treebank.

Figure 2: Dependency tree by PENN2MALT.
Conversion for BulTreeBank

There four conversion from HPSG-based Treebank to dependency Treebank:
• Marinov 2005 – their own head percolation table and dependency labels – BulTreeBank-M
• Chanev 2006 – their own head percolation table and dependency labels – BulTreeBank-C
• Simov and Osenova 2006 – CoNLL-X shared task 2006
• Simov and Osenova 2014 – Universal Dependancy
BulTreeBank in CoNLL-X Format

• Instead of head percolation table the actual annotation of head is used
• Creation of BulTreeBank specific dependency tagset
• The dependency labels depends on the type of headed and non-headed phrases
<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjunct</td>
<td>Adjunct (optional verbal argument)</td>
</tr>
<tr>
<td>clitic</td>
<td>Clitic form</td>
</tr>
<tr>
<td>comp</td>
<td>Complement (arguments of: non-verbal heads, non-finite verbal heads, copula)</td>
</tr>
<tr>
<td>conj</td>
<td>Conjunction in coordination</td>
</tr>
<tr>
<td>conjarg</td>
<td>Argument (second, third, …) of coordination</td>
</tr>
<tr>
<td>indobj</td>
<td>Indirect Object (indirect argument of a non-auxiliary verbal head)</td>
</tr>
<tr>
<td>marked</td>
<td>Marked (clauses, introduced by a subordinator)</td>
</tr>
<tr>
<td>mod</td>
<td>Modifier (dependants which modify nouns, adjectives, adverbs)</td>
</tr>
<tr>
<td>obj</td>
<td>Object (direct argument of a non-auxiliary verbal head)</td>
</tr>
<tr>
<td>pragadjunct</td>
<td>Pragmatic adjunct</td>
</tr>
<tr>
<td>prepcomp</td>
<td>Complement of preposition</td>
</tr>
<tr>
<td>punct</td>
<td>Punctuation</td>
</tr>
<tr>
<td>subj</td>
<td>Subject</td>
</tr>
<tr>
<td>xadjunct</td>
<td>Clausal adjunct</td>
</tr>
<tr>
<td>xcomp</td>
<td>Clausal complement</td>
</tr>
<tr>
<td>xmod</td>
<td>Clausal modifier</td>
</tr>
<tr>
<td>xprepcomp</td>
<td>Clausal complement of preposition</td>
</tr>
<tr>
<td>xsubj</td>
<td>Clausal subject</td>
</tr>
</tbody>
</table>
In my notebook I have recorded the following lines.
Who does not have anything, promises easily.

Който няма, лесно обещава.
Problematic Cases

- **NN construction**: чаша вода (glass of water)
  - Head initial or head final
- **Coordination**
  - Determination of the head word
- **Substantivation**
  - Category change
- **Ellipses**
  - Empty elements
Coordination

He has written quite a few short novels and dramas.

30th European Summer School in Logic, Language and Information (6 August – 17 August 2018)
Current Status

Universal Dependencies (UD) is a framework:

- for cross-linguistically consistent grammatical annotation, and
- an open community effort
- with over 200 contributors producing
- more than 100 treebanks
- in over 60 languages.
UD: Overview (1)

Universal Dependencies (UD) is a project that is developing cross-linguistically consistent treebank annotation for many languages, with the goal of facilitating:

- multilingual parser development,
- cross-lingual learning, and
- parsing research from a language typology perspective
UD: Overview (2)

The general philosophy is:

• To provide a universal inventory of categories and guidelines to facilitate consistent annotation of similar constructions across languages, while allowing language-specific extensions when necessary
What is Needed for UD to be Successful?

The secret to understanding the design and current success of UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD needs to be satisfactory on linguistic analysis grounds for individual languages.
2. UD needs to be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
What is Needed for UD to be Successful? (2)

3. UD must be suitable for rapid, consistent annotation by a human annotator.
4. UD must be suitable for computer parsing with high accuracy.
5. UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. We refer to this as seeking a habitable design, and it leads us to favor traditional grammar notions and terminology.
What is Needed for UD to be Successful? (3)

6. UD must support well downstream language understanding tasks (relation extraction, reading comprehension, machine translation, …).
History

The annotation scheme is based on an evolution of:

• (universal) Stanford dependencies (de Marneffe et al., 2006, 2008, 2014),
• Google universal part-of-speech tags (Petrov et al., 2012), and
• Interset interlingua for morphosyntactic tagsets (Zeman, 2008)
• The first attempt to combine Stanford dependencies and Google universal tags was the Universal Dependency Treebank (UDT) project (McDonald et al., 2013) - released treebanks for 6 languages in 2013 and 11 languages in 2014
History

• The first proposal for incorporating morphology was made by Tsarfaty (2013).
• The second version of HamleDT (Rosa et al., 2014) provided Stanford/Google annotation for 30 languages in 2014. This was followed by the development of universal Stanford dependencies (USD) (de Marneffe et al., 2014).
History

- The dependency representation of UD evolves out of Stanford Dependencies (SD), which itself follows ideas of grammatical relations-focused description that can be found in many linguistic frameworks.
- That is, it is centrally organized around notions of subject, object, clausal complement, noun determiner, noun modifier, etc.
Related Work

- *Rosa et al.* (2014) where 30 treebanks have been harmonized into a common Prague Dependency style, and then converted into Stanford Dependencies
- *Sanguinetti and Bosco* (2014) and *Bosco and Sanguinetti* (2014) on conversion of the parallel treebank ParTUT (Italian, English, French) into Stanford dependencies
- *Lipenkova and Soucek* (2014) on Russian dependency treebank into Stanford dependencies
- Conversion of the Swedish treebank (*Nivre*, 2014) and the Finnish treebank (*Pyysalo et al.*, 2015) to UD
Unlabeled Dependencies
Labeled Dependencies
The Choice of Dependencies

- Dependency Grammar theories are based on the observation that many syntactic relationships can be characterized as asymmetric, binary relations between head and modifier words. (Tesnière 1959, Sgall et al. 1986, …)
  - If you learned sentence diagramming in grade school (Reed & Kellogg 1877), that is a form of dependency grammar!
  - Not all constructions fit cleanly (coordination, relative clauses, …); different theories have different solutions. Labeling the dependencies can clarify the nature of the relationship.
The Choice of Dependencies

• While constituency grammars work well for “well-behaved” languages like English, Turkish and other languages introduce complications.
• Because dependency parses are structurally simpler, they are computationally easier to produce. (Faster parsers!)
• Syntactic dependencies are not too far from semantic dependencies, useful for many applications.
Cross-Lingual Similarity
## Universal POS Categories

<table>
<thead>
<tr>
<th>Open class words</th>
<th>Closed class words</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJ</td>
<td>ADP</td>
<td>PUNCT</td>
</tr>
<tr>
<td>ADV</td>
<td>AUX</td>
<td>SYM</td>
</tr>
<tr>
<td>INTJ</td>
<td>CCONJ</td>
<td></td>
</tr>
<tr>
<td>NOUN</td>
<td>DET</td>
<td></td>
</tr>
<tr>
<td>PROPN</td>
<td>NUM</td>
<td></td>
</tr>
<tr>
<td>VERB</td>
<td>PART</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PRON</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SCONJ</td>
<td>X</td>
</tr>
</tbody>
</table>
Universal Grammatical Features

<table>
<thead>
<tr>
<th>Lexical features</th>
<th>Inflectional features</th>
</tr>
</thead>
<tbody>
<tr>
<td>PronType</td>
<td>Nominal*</td>
</tr>
<tr>
<td>NumType</td>
<td>Gender</td>
</tr>
<tr>
<td>Poss</td>
<td>Animacy</td>
</tr>
<tr>
<td>Reflex</td>
<td>NounClass</td>
</tr>
<tr>
<td>Foreign</td>
<td>Number</td>
</tr>
<tr>
<td>Abbr</td>
<td>Case</td>
</tr>
<tr>
<td></td>
<td>Effective</td>
</tr>
<tr>
<td></td>
<td>Degree</td>
</tr>
<tr>
<td></td>
<td>Polarity</td>
</tr>
<tr>
<td></td>
<td>Person</td>
</tr>
<tr>
<td></td>
<td>Polite</td>
</tr>
<tr>
<td></td>
<td>Clusivity</td>
</tr>
<tr>
<td></td>
<td>Verbal*</td>
</tr>
<tr>
<td></td>
<td>VerbForm</td>
</tr>
<tr>
<td></td>
<td>Mood</td>
</tr>
<tr>
<td></td>
<td>Tense</td>
</tr>
<tr>
<td></td>
<td>Aspect</td>
</tr>
<tr>
<td></td>
<td>Voice</td>
</tr>
<tr>
<td></td>
<td>Evident</td>
</tr>
<tr>
<td></td>
<td>Polarity</td>
</tr>
<tr>
<td></td>
<td>Person</td>
</tr>
<tr>
<td></td>
<td>Polite</td>
</tr>
<tr>
<td></td>
<td>Clusivity</td>
</tr>
</tbody>
</table>
## Dependency Relations

<table>
<thead>
<tr>
<th></th>
<th>Nominals</th>
<th>Clauses</th>
<th>Modifier words</th>
<th>Function Words</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Core arguments</strong></td>
<td>nsubj</td>
<td>csubj</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>obj</td>
<td>ccomp</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>obj</td>
<td>xcomp</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Non-core dependents</strong></td>
<td>obl</td>
<td>advcl</td>
<td>advmod*</td>
<td>aux</td>
</tr>
<tr>
<td></td>
<td>vocative</td>
<td></td>
<td>discourse</td>
<td>cop</td>
</tr>
<tr>
<td></td>
<td>expl</td>
<td></td>
<td></td>
<td>mark</td>
</tr>
<tr>
<td></td>
<td>dislocated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Nominal dependents</strong></td>
<td>nmod</td>
<td>acl</td>
<td>amod</td>
<td>det</td>
</tr>
<tr>
<td></td>
<td>appos</td>
<td></td>
<td></td>
<td>clf</td>
</tr>
<tr>
<td></td>
<td>nummod</td>
<td></td>
<td></td>
<td>case</td>
</tr>
</tbody>
</table>
Not Dependency Relations in the Narrow Sense

<table>
<thead>
<tr>
<th>Coordination</th>
<th>MWE</th>
<th>Loose</th>
<th>Special</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>conj, cc</td>
<td>fixed</td>
<td>list</td>
<td>orphan, goeswith</td>
<td>punct, root, dep</td>
</tr>
<tr>
<td></td>
<td>flat</td>
<td>parataxis</td>
<td>reparandum</td>
<td></td>
</tr>
<tr>
<td></td>
<td>compound</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Subject, Object, Oblique

- She left a note to him.
- She left a note on the table.
Core Arguments
Core Arguments

<table>
<thead>
<tr>
<th>subjects</th>
<th>objects</th>
<th>obliques</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsubj nominal subject</td>
<td>obj direct object</td>
<td>obl case-marked noun</td>
</tr>
<tr>
<td>nsubj:pass nominal subject of passive</td>
<td>iobj indirect object</td>
<td>obl:agent passive by argument</td>
</tr>
<tr>
<td>csubj clausal subject</td>
<td>advmod modifying adverb</td>
<td>obl:tm mod temporal noun (adverbial or case-marked)</td>
</tr>
<tr>
<td>csubj:pass clausal subject of passive</td>
<td>expl expletive subject</td>
<td></td>
</tr>
</tbody>
</table>

Here are some examples:
- There is a ghost in the room.
- It is clear that we should decline.
Auxiliaries, Copulas, Case

Remember: these are **function** words, so they modify content words.

- **aux**: auxiliary
- **cop**: copula
- **case**: preposition or case clitic modifying a nominal

**aux**:pass passive auxiliary (form of *be* or *get*)

- **nsubj**
- **aux**
- **cop**

- **infl**

`she has been happy`

- **nsubj**
- **cop**

`she is my mother`

- **nsubj**
- **cop**
- **case**

`she is in the kitchen`

- **expl**
- **nsubj**
- **obl**
- **case**

`there is food in the kitchen`
Adjectives, Determiners, Nominal Modifiers
Compounds, Flat names, Fixed Expressions

- **compound**
  - **compound:p** verb particle

- **compound:s** serial verb construction

- **flat** names without head-modifier structure

- **fixed** fixed grammatical expressions

With **fixed** and **flat**, the first word heads all other words in the expression.

Examples:
- **compound:p**
  - I bought a computer disk
  - We had a nice time in spite of the rain

- **compound:s**
  - I drive enclosure

- **flat**
  - Martin Luther King had a dream
Coordination
Complement Clauses

- **ccomp**: clausal complement
- **mark**: subordinator, complementizer, or infinitive marker
- **xcomp**: a predicate's clausal (or predicate A/N) complement that shares an argument with the matrix predicate

Examples:
- He says you like to swim
- He says that you like to swim
- The boss said to start digging
- Please let us know
- Sue persuaded Fred to accept the job.
Modifier Clauses

**advcl** adverbial clause (e.g. expressing time, purpose, reason, condition...)

**acl** adjectival clause

**acl:relcl** relative clause

- The accident happened as night was falling
- He talked to him in order to secure the account
- If you know who did it, you should tell the teacher
- The issues as he sees them
- Sam spent everything he had
- Sam spent everything that he had
Discourse

- **vocative**: address
- **dislocated**: topicalized noun phrase
- **discourse**: single word functioning as an interjection, filler, or similar conversational marker

Diagram showing examples of discourse in sentences:
- "Guys, take it easy!
- "Yes, we should apply for membership:
- "Anyway, the mezza luna"
- "Please join our growing family!"

Diagram includes various linguistic elements such as adverbs, nouns, verbs, and punctuation marks.
Enhanced Dependencies

- They include the following phenomena:
  - Null nodes for elided predicates.
  - Propagation of conjuncts.
  - Additional subject relations for control and raising constructions.
  - Arguments of passives (and other valency-changing constructions).
  - Coreference in relative clause constructions.
  - Modifier labels that contain the preposition or other case-marking information.
BulTreeBank in UD: BulTreeBank-Original

- The original BulTreeBank (Simov et al., 2004; Simov and Osenova, 2003) comprises 214,000 tokens, which form a little more than 15,000 sentences.

- It is in XML format and is based on HPSG.

- The syntactic structure is presented through a set of constituents with head-dependant markings.
Domains (Genres)

- The domains in the treebanks are as follows:
  - Bulgarian newspapers (81 %)
  - fiction (16 %)
  - administrative documents (3 %)
BulTreeBank in UD: BulTreeBank in Conversions

• First ‘glocalization’ happened in 2006, when it was converted into the shared CoNLL dependency format – (Chanev et al., 2006) and (Chanev et al., 2007)
• Alternative versions of BulTreeBank exist in two other popular formats:
  • PennTreebank (Ghayoomi et al., 2014) and
  • Stanford Dependencies (Rosa et al., 2014)
BulTreeBank in UD: Status

- The UD_Bulgarian-BTB:
  - consists of 156,149 tokens (11,138 sentences). This subset of BulTreeBank excludes:
    - ellipses
    - some rare phenomena
  - The conversion was done semi-automatically by Kiril Simov, with the application of set of rules and constraints for consistency.
  - The rest of the sentences will be converted for the next releases. The original version is freely available for research upon request.
Our Linguistic Approach to UD

- Data release status: UD v1.1 (Agic et al., 2015) together with other 17 languages. Its size is 125 000 tokens, which constitute half of the data.
- We postponed the addition of language specific features for the next stage
- Mappings:
  - The morphological mapping includes parts-of-speech and their lexical as well as inflectional features
  - The syntactic mapping focuses on dependency relations.
Morphology (1)

• From the original tagset to the UD tagset:
  • **Direct Mapping**: subordinators and conjunctions, adjectives, prepositions
  • **Division of one POS into more parts-of-speech**:  
    • From morphological approach to syntactic (functional)  
      • the PRONOUNS are split in three groups: DET, PRON and ADV  
      • Numerals also divide between the groups of ADJ, ADV and NUM
Morphology (2)

• The verbs are divided into the groups VERB (main verbs, copulas and modals, participles that are part of verb forms), AUX (auxiliaries), ADJ (participles with attributive usages).

• **Changing the POS:**
  • the transition of the affirmative and negative particles to the group of INTJ (interjections)
  • all the pronouns that went to DET group, also changed their POS label.
Syntax (1)

• **Direct transfer relations:**
  - *dobj, iobj, nsubj, csubj*
  - Distinction between the relations *aux* and *copula* is directly derived from the original annotation

• **Non-direct relations:**
  - Division of our original complement clauses (CLDA, CLCHE, etc.) into control (*xcomp*) and non-control ones (*ccomp*)
Syntax (2)

• Division of our head-adjunct nominal phrase (NPA) into several relations depending on the non-head sister: nummod (the non-head sister is numeral), amod (the non-head sister is adjective), det (the non-head sister is determiner).

• ‘Floating’ relations:
  • The case of encoding the question particle ‘li’ in Bulgarian as relation *discourse*, but
  • there are also other options, such as *aux*, *expl* or *mark*. 
Аз съм, ако искаш да знаеш, в най-решителния завой на живота си.
Conversion Procedure (1)

- MWEs were analyzed syntactically for tokenization reasons
- The parts-of-speech together with the relevant grammatical features were converted automatically through pre-defined mappings
- The syntactic relations required more work. Part of them were converted automatically, while part of them needed human intervention
Conversion Procedure (2)

• More specific rules are needed in some combinations of constituents:
  • in NPs of type NN the head might be the first or the second noun depending on the semantics of the phrase
  • Coordinations originally have been considered to be non-headed phrases. Thus, they need special treatment.
Conversion Procedure (3): Rule-based

- The rules are of two kinds:
  - lexical head identifier moving up the constituent tree; and
  - relation assignment for a constituent node of the dependent child when all children of the parent node have lexical identifiers.
- Applied recursively to the root
Example

BG: твърде висок зелен стол
EN: too tall green chair

[NPA [APA too tall] [NPA green chair]]

NPA -> APA id1 NPA id2
NPA id2 -> APA id1 amod NPA id2
Preliminary Experiments: Set-up

- **Training resource**: BTB-UD
- **Approach**: 10-fold cross validation
- **Task**: POS tagging and Dependency Parsing
- **Tool**: MATE
- **State-of-the-art**:
  - POS tagging (680 tags): 97.98 % accuracy (Georgiev et al., 2012)
  - Dependency parsing: LAS – 89.14 % and UAS – 92.45 % (Simova et al., 2014), using an ensemble model
Preliminary Experiments (on 535 tags): Results

<table>
<thead>
<tr>
<th>Task</th>
<th>Accuracy</th>
<th>LAS</th>
<th>UAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>POS Tagging</td>
<td>96.89 %</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Mor. Tagging</td>
<td>98.50 %</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Dep. Parsing</td>
<td>–</td>
<td>83.50 %</td>
<td>88.08 %</td>
</tr>
</tbody>
</table>
What can be Released (1)

- Corrections with the use of *copula relation*: it was extended and applied to sentences that consist of the auxiliary verb ‘to be’, a noun and a subordinate sentence. In this case noun is the root, auxiliary verb is the dependent, attached with a cop relation, and the subordinate sentence is the subject.
What can be Released (2)

- appositions (the wrongly annotated vocatives and appositions were corrected as nmod)
- xcomp (the ccomp relations were checked for control structures and re-annotated as xcomp)
- nsubjpass and csubjpass in se-passives were annotated
- cop relation direction was corrected
Transfer: Null Nodes for Elided Predicates

- In BTB such predicates are introduced as V-Elip (262 inst) or VD-Elip (255 inst)
- V-Elip is the more straightforward one, while VD-Elip considers also cases of VP-ellipsis and copula ellipsis
- VD-Elip provides discourse labels with the meaning that it is difficult to identify the type (let alone the form) of the missing element(s). These difficult cases can be processed only manually
Towards Enhanced Dependencies: Original Ellipsis
Transfer: Propagation of Conjuncts

• We have to rely on the implicit but straightforward information
• In BTB the coordination phrases are considered head-less and thus – flat
• Thus *amod* relations can be established on the base of the morphosyntactic and lexical information coming from the elements of the coordination phrase. The same holds for the core/non-core arguments
Example
Example

The voice was singing with tired and flat sorrow.
Transfer: Additional Subject Relations for Control and Raising Constructions

• In BTB there is a relation between the unexpressed subject of the embedded verb (pro-ss) and the subject of the matrix clause

• Thus, the nsubj relation between the subject of the main verb and the embedded verb can be established rather easily

• Just the pro-ss element has to be substituted with nsubj and to be moved on the verb itself
Towards Enhanced Dependencies: Original Pro-drop and Discontinuity
Transfer: Arguments of Passives (and Other Valency-changing Constructions)

• In BTB there are no special markings of these arguments
• Some of them:
  • Can be derived automatically (such as the participle passive due to its special morphological form), and
  • Some of them are not trivial, such as the se-passives (being formed with the originally reflexive accusative clitic se attached to the tensed verb form, since they are ambiguous across types of voice as well as markers of intransitive/detransitivised verbs
• In the present UD version these labels are already available
Transfer: Coreference in Relative Clause Constructions

• The representation in BTB provides constituency information on the modified by the relative element.

• Thus, the ref relation from this modified element to the relative pronoun can be established automatically.

• As a result, also the nsubj relation can be established between the modified nominal head and the predicate in the embedded clause.
Example
Example
Transfer: Modifier Labels that Contain the Preposition or Other Case-marking Information

- Since Bulgarian is analytic language, the non-core or nominal dependants (nmod, obl, acl and advcl) would have labels with propagated prepositions.
- This step can be done automatically.
Conclusions

- Conversion was done in mainly automatic, rule-based way
- Some phenomena require special attention
- The main challenge is the proper handling of the language universal and language specific phenomena at a minimal linguistic and data model loss
- The most important perspective is the ultimate goal of having comparable syntactically annotated resources for many languages that would serve better for various NLP tasks
References


References

