Phenomenon Description Analyzer (PDA)

By Kevin O'Brien

1. Abstract

The Phenomenon Description Analyzer (PDA) has been researched and created in order to allow people without a background in simulations to have a description of their phenomenon separated into the components of an agent-based model (ABM). In order to implement this technology, IBM's Watson Natural Language Understanding (NLU) and Knowledge Studio were used alongside each other in order to extract entities from unstructured descriptions of phenomenons.

2. Process

Natural language processing (NLP) is a subcategory of Artificial Intelligence (AI) focused on allowing the computer to process, understand and communicate with natural language. This is achieved through the combination of syntax, semantics, and pragmatics alongside machine learning algorithms. Natural language understanding (NLU) is a component of NLP which is used to handle unstructured inputs that are poorly defined with flexible rules and convert them into a structured form that a machine can understand and act upon (Dahlgren & Stabler 1997).

The process of converting unstructured natural language into structured data consists of multiple different steps. These steps must be done in sequential order and include tokenization, parts of speech tagging, parts of speech chunking, and named entity recognition. The listed steps are what allows the computer to understand the text more efficiently than rule-based algorithm alone (Dahlgren & Stabler 1997).

Tokenization is the process of taking the body of unstructured data and breaking every word into a token. This is done in order to tag each token with a part of speech. Tagging each token with a part of speech allows the computer to gain some insight into what it is reading. Once the tagging is completed, the tags are chunked into the syntactic parse tree(**figure 1.0**) also known as parts of speech chunking. This is what allows the computer to understand how sentences are structured and what they imply. Once this has been complete you can create named entities such as a person, place, or organization. To do this the user informs the computer where the entity is within the text body and the entities name. Once this has been done for the entire document(s) the algorithm is then trained. When given an untrained document the computer is able to determine that named entity due to its parts of speech and placement within the parse tree in relation to your training.

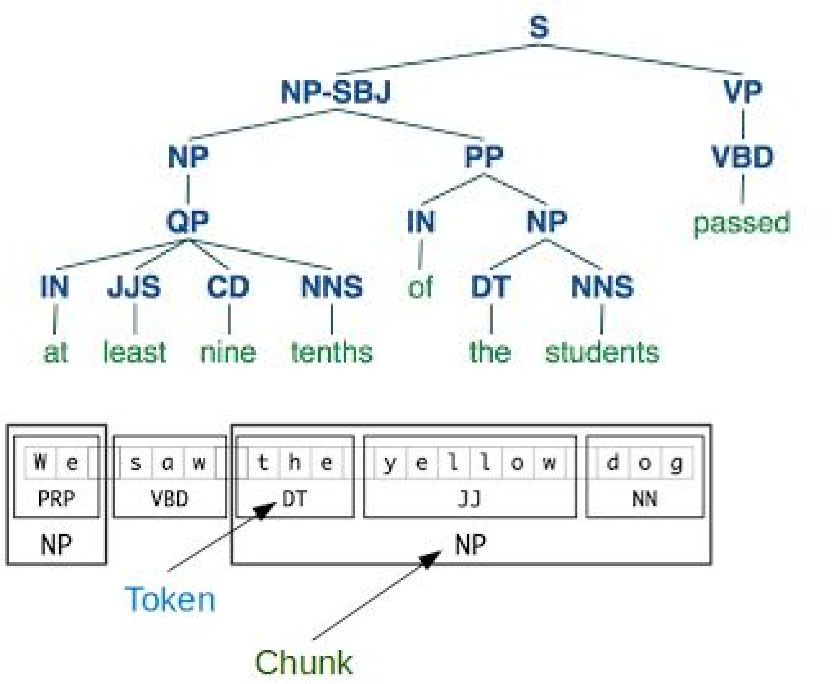


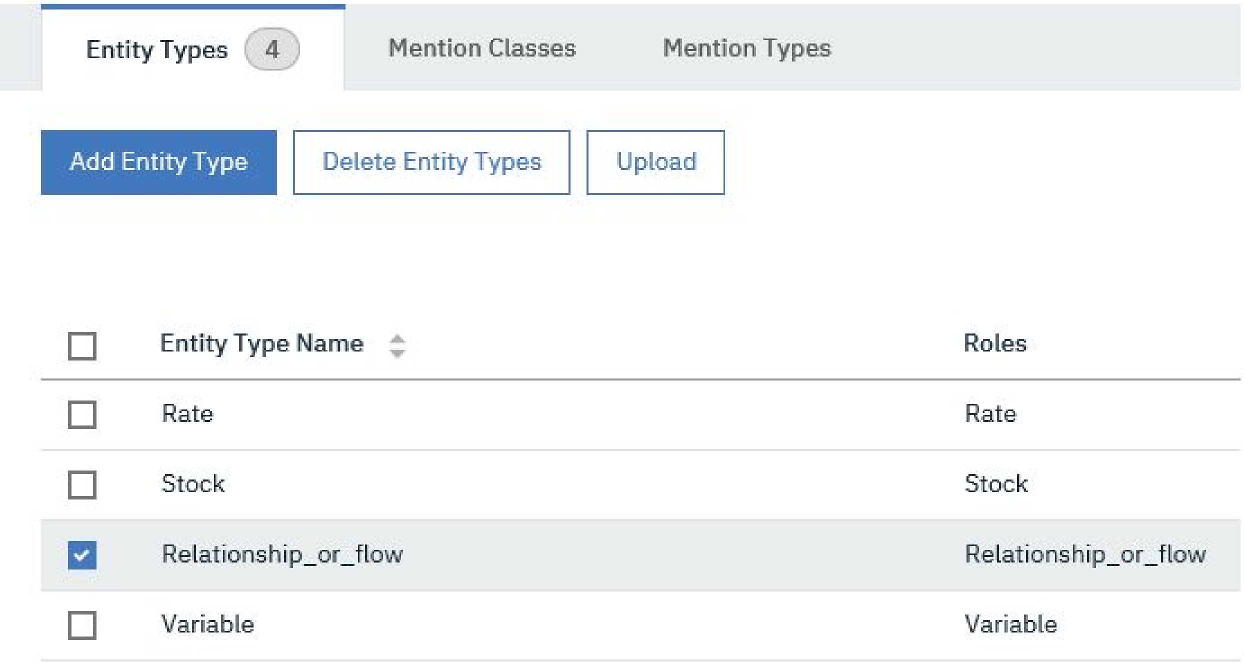
figure 2.1 (an example of syntactic parsing)

3. Tool

The Artificial intelligence used are IBM's Watson's sub-products; Natural Language Understanding (NLU) and Knowledge Studio. The NLU is capable of finding keywords such as sentiment, emotion, default entities, and concepts within a sentence or body of text.

Watson Knowledge Studio allows a user to fill a corpus and train a model to look for named entities that the user names and defines such as the ones in **Figure 3.1**. Watsons NLU uses all of the steps named above all in one tool allowing anyone to create their own model without code.

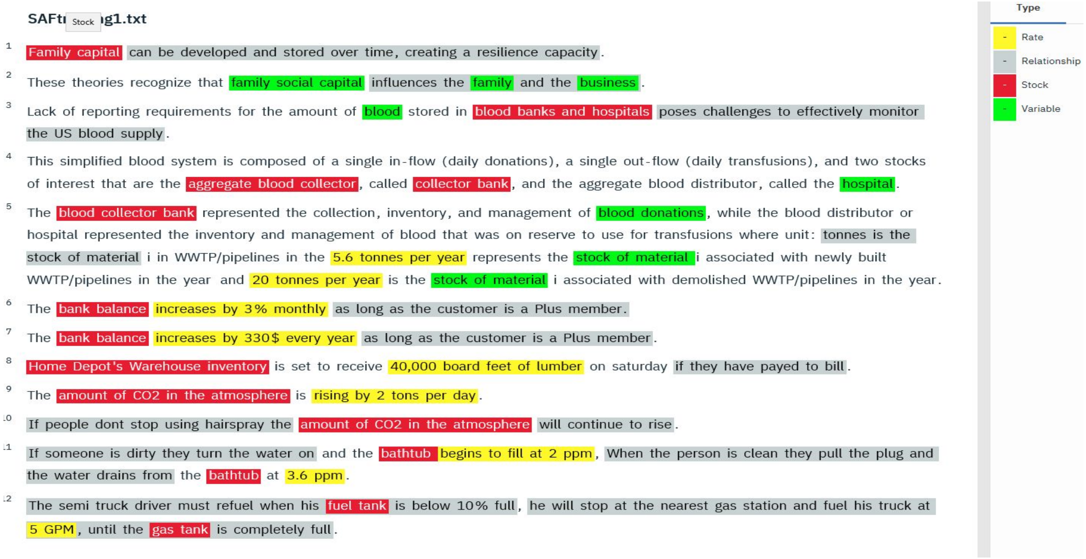
In order to train the model, the user must define the entities within the corpus by annotating each document. Annotating refers to selecting the results the user would like to see from a document through highlighting. Once trained, the user receives a model id which can be imported into the NLU, allowing it to search any description for the entities the user defined.



□	Entity Type Name	Roles
□	Rate	Rate
□	Stock	Stock
☑	Relationship_or_flow	Relationship_or_flow
□	Variable	Variable

Figure 3.1 (Entities tab where the user can create their custom entities)

The corpus for the model used in finding agent-based related entities had its corpus filled with descriptions from the Journal of Artificial Societies and Social Simulation (JASSS). Below, **Figure 3.2** displays an example of how documents are annotated.



1 Family capital can be developed and stored over time, creating a resilience capacity.

2 These theories recognize that family social capital influences the family and the business.

3 Lack of reporting requirements for the amount of blood stored in blood banks and hospitals poses challenges to effectively monitor the US blood supply.

4 This simplified blood system is composed of a single in-flow (daily donations), a single out-flow (daily transfusions), and two stocks of interest that are the aggregate blood collector, called collector bank, and the aggregate blood distributor, called the hospital.

5 The blood collector bank represented the collection, inventory, and management of blood donations, while the blood distributor or hospital represented the inventory and management of blood that was on reserve to use for transfusions where unit: tonnes is the stock of material i in WWTP/pipelines in the 5.6 tonnes per year represents the stock of material i associated with newly built WWTP/pipelines in the year and 20 tonnes per year is the stock of material i associated with demolished WWTP/pipelines in the year.

6 The bank balance increases by 3% monthly as long as the customer is a Plus member.

7 The bank balance increases by 330$ every year as long as the customer is a Plus member.

8 Home Depot's Warehouse inventory is set to receive 40,000 board feet of lumber on saturday if they have payed to bill.

9 The amount of CO2 in the atmosphere is rising by 2 tons per day.

10 If people dont stop using hairspray the amount of CO2 in the atmosphere will continue to rise.

11 If someone is dirty they turn the water on and the bathtub begins to fill at 2 ppm, When the person is clean they pull the plug and the water drains from the bathtub at 3.6 ppm.

12 The semi truck driver must refuel when his fuel tank is below 10% full, he will stop at the nearest gas station and fuel his truck at 5 GPM, until the gas tank is completely full.

Figure 3.2 (Example of the annotation process)

After annotating is complete, the user must confirm the documents are correct and then training begins. Training can take anywhere from 5 minutes to multiple hours depending on the number of documents added and their size.

The Knowledge Studio allows the user to run new descriptions against it and have it pre-annotate the document automatically. Then the user can correct anything wrong with the annotation produced by the model (same way as **Figure 3.2)** and add it to the corpus. Once it's added it can be submitted and trained again. It is easy to return to a model and continue to upgrade its accuracy.

4. Training Methods

Every time a model was trained, it went through the same series of tests with a few exceptions depending on what was trying to be improved.

Before a model can be trained, the user selects which type of entities the model should be able to analyze. Rules must be set in place in order to ensure that the user's annotations stay consistent with their corresponding entity. To determine these rules it's important to take a look at what parts of speech that entity is and make sure to follow that during annotation. If your entity is multiple words long a majority of the time then the user should identify the parts of speech phrase and follow that instead.

In the case of the PDA, we are looking for agents, attributes, rules, variables, and relationships. The more entities a model looks for the more inaccurate it becomes. If the model finds a string and determines it is a certain entity, the string will not be found as another entity. This leads to problems with agent rules in our case, which are generally multiple words long and also contain agents and attributes within their string. If it is determined to be a rule by the PDA, the agents and attributes will not be returned in the results.

The way to combat this issue is by creating a separate model for rules. Agents and attributes can stay in one model because they are typically short strings of 1-2 words and are completely different parts of speech (i.e noun, adjective).

Documents selected for training and testing should be similar to what the model will be used for. For our model, we used descriptions of phenomenons from the JASSS articles.

The documents used are all descriptions of phenomenons but are all vastly different topics to ensure a diverse corpus. This will allow more accuracy with the limitless possible descriptions(Sabou 2014).

Documents that contain an upwards of 4-5 lines where the user decides those lines do not contain an entity should be removed from the text all together as this can weaken results.

Before annotation can commence, the user must determine what the rules are for the entities. In the PDA we used the following rules for ABM portion:

> -Agents have attributes and also perform tasks, actions, and decisions.
> **ex**: (person, group, animal)

-Attributes are related to agents and can increase, decrease, or change in some way. **ex**: (age, wage, weight, productivity)

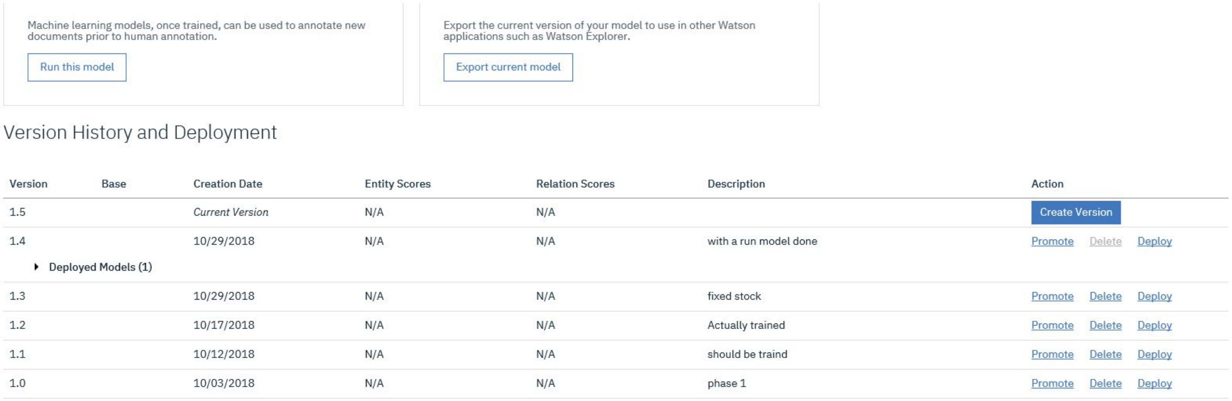
-Rules contain an action verb and usually relate the agent to an attribute or vice versa. **ex**: (Societies with more power will often attack neighbors; People who work have a higher income.)

It is best when beginning training to use 3-4 documents every time the model is upgraded. After training, the accuracy should be tested against an untrained document and the results should be reviewed.

After checking the results, if expectations are not reached, the new training documents can then be pre-annotated by the model and the user can correct the results. Then it is submitted and trained again, this process is repeated until expectations are reached.

By annotating and training 3-4 documents at a time and following the methods above, the user can catch any mistakes that start to occur and determine why they happened.

If the model seems worse than before the training, it is easy to go back to an earlier version as seen in **Figure 4.1** below.



Machine learning models, once trained, can be used to annotate new documents prior to human annotation.

Run this model

Export the current version of your model to use in other Watson applications such as Watson Explorer.

Export current model

Version History and Deployment

Version	Base	Creation Date	Entity Scores	Relation Scores	Description	Action
1.5		*Current Version*	N/A	N/A		Create Version
1.4		10/29/2018	N/A	N/A	with a run model done	Promote Delete Deploy
▸ Deployed Models (1)						
1.3		10/29/2018	N/A	N/A	fixed stock	Promote Delete Deploy
1.2		10/17/2018	N/A	N/A	Actually trained	Promote Delete Deploy
1.1		10/12/2018	N/A	N/A	should be traind	Promote Delete Deploy
1.0		10/03/2018	N/A	N/A	phase 1	Promote Delete Deploy

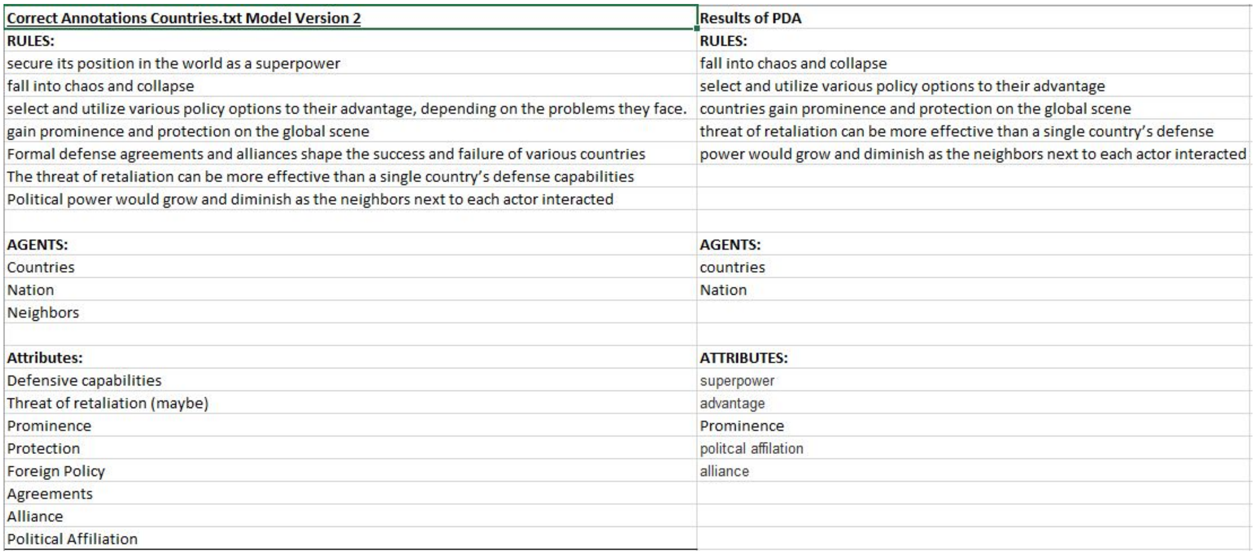
Figure 4.1 (The image above shows an example of the versions of a single model)

5. Testing Results

Testing was done by comparing the desired results of testing documents to the results from the PDA as shown in **Figures 5.1, 5.2**. Desired results were obtained by manually annotating the document.

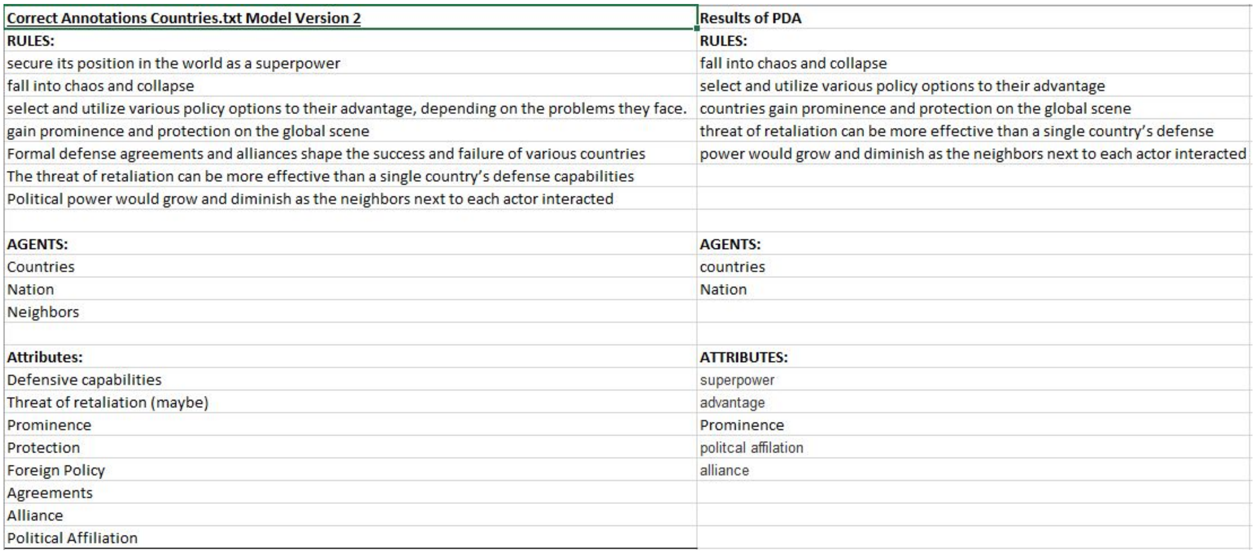
In **Figure 5.1** below, results from version one of a model are shown. The document was not used in training and was saved to compare to results from the PDA multiple times until the results are accurate (Sabou 2014). After it is accurate enough, it should be added to the training documents that will be annotated.

When observing the results, the user should look for trends in order to determine what could be causing incorrect results. Usually, the first version of a model will not be very accurate. This is due to a lack of diversity in the training documents.



Correct Annotations Countries.txt Model Version 2	Results of PDA
RULES:	RULES:
secure its position in the world as a superpower	fall into chaos and collapse
fall into chaos and collapse	select and utilize various policy options to their advantage
select and utilize various policy options to their advantage, depending on the problems they face.	countries gain prominence and protection on the global scene
gain prominence and protection on the global scene	threat of retaliation can be more effective than a single country’s defense
Formal defense agreements and alliances shape the success and failure of various countries	power would grow and diminish as the neighbors next to each actor interacted
The threat of retaliation can be more effective than a single country’s defense capabilities	
Political power would grow and diminish as the neighbors next to each actor interacted	
AGENTS:	AGENTS:
Countries	countries
Nation	Nation
Neighbors	
Attributes:	ATTRIBUTES:
Defensive capabilities	superpower
Threat of retaliation (maybe)	advantage
Prominence	Prominence
Protection	politcal affilation
Foreign Policy	alliance
Agreements	
Alliance	
Political Affiliation	

Figure 5.1 (Results from version 1 test against training document Countries.txt)



Correct Annotations Countries.txt Model Version 2	Results of PDA
RULES:	RULES:
secure its position in the world as a superpower	fall into chaos and collapse
fall into chaos and collapse	select and utilize various policy options to their advantage
select and utilize various policy options to their advantage, depending on the problems they face.	countries gain prominence and protection on the global scene
gain prominence and protection on the global scene	threat of retaliation can be more effective than a single country’s defense
Formal defense agreements and alliances shape the success and failure of various countries	power would grow and diminish as the neighbors next to each actor interacted
The threat of retaliation can be more effective than a single country’s defense capabilities	
Political power would grow and diminish as the neighbors next to each actor interacted	
AGENTS:	AGENTS:
Countries	countries
Nation	Nation
Neighbors	
Attributes:	ATTRIBUTES:
Defensive capabilities	superpower
Threat of retaliation (maybe)	advantage
Prominence	Prominence
Protection	politcal affilation
Foreign Policy	alliance
Agreements	
Alliance	
Political Affiliation	

Figure 5.2 (results from version 2 test against training document Countries.txt)

Results were then compared to version 2 after another 3 documents were annotated and training was complete.

It is important not to switch testing documents until the model has reached an 85% or higher ratio. The user needs to be able to accurately track result progression which is impossible if the testing document is switched. The training methods above are what allowed for visible evidence of accuracy progression.

6. Observations

In the case of agent rules where they typically contain an agent and/or attribute, it looks best to run the agent and attribute model against the results from the rules model. This is possible by separating the models and may allow for a confirmation on the agents returned.

NLU's built-in "keyword" that is returned when a document is analyzed does a good job of breaking a description down into all of its different pieces and is comparable to the custom entity Variable from the SD model in Metaphr. It seems as though keywords could replace Variable or be used alongside It. Testing was done with the NLU's default "entities" which extract people, companies, organizations, cities, geographic features, and other information from the content and was found that it can be helpful to the user.

However, the default "entities" option used for finding agents does not work, as it will only return results of people or places if they are referred to by name. If in a description, the agent is "people" or "families" entities will return no results.

The original intention was to have two separate analyzers depending on the user's model description either as agent-based or system dynamics, but it was found that together, they help clarify how the entire model flows and is related.

NLU and Knowledge Studio are not capable of returning the sentence in which an entity was found. This feature would help with showing the relation and flow of a description, but more will have to be done before the PDA is capable of that. A python script may be able to achieve this for the PDA.

The PDA can be updated at any time with more models that can be added to find new types of entities or updated models for more accuracy.

References

Dahlgren, K. Edward, S. " Natural Language Understanding Systems " Intelligent Text Processing Inc Cognition Technologies Inc, 1997.

Sabou, M., Bontcheva, K., Derczynski, L., & Scharl, A. (2014). Corpus annotation through crowdsourcing: Towards best practice guidelines. In *Proceedings of the 9th international conference on language resources and evaluation (LREC'14)* (pp. 859–866).