Title, author list not final. Page limit: 5 + 2 bib

A Classifier for Communicative Functions of Definiteness

Abstract

Definiteness is a nonhomogeneous category across languages expressing various *communicative functions* (types of semantic, pragmatic, and discourse–related information). Languages differ in the grain-size and inventory of communicative functions that are expressed as well as the syntactic means for expressing them. **?** presented an annotation scheme for the communicative functions of definiteness and released an annotated corpus in English and Hindi. This paper reports on a classifier for English that uses lexical, morphological, and syntactic features to predict communicative functions. The benefits of this work are twofold: linguistically, the classifier's features and weights model the *grammaticalization* of definiteness in English, some of which is obvious (e.g., *the* and *a*) and some of which is not obvious [$^{A}_{B}$ (include an example from our findings)]. Computationally, it presents a framework to predict *semantic and pragmatic* communicative functions of definiteness which, unlike lexical and morphosyntactic features, are preserved in translation. This work may also be useful in tracking noun phrase reference for information extraction, entity tracking, and other semantic processing tasks.

1 Introduction

Languages display a vast range of variation with respect to the form and meaning of definiteness. For example, while languages like English make use of definite and indefinite articles to distinguish between the discourse status of various entities (*the car* vs. *a car*), many other languages, such as Czech, Hindi, Indonesian, and Russian, do not have articles (although they do have demonstrative determiners). Some languages such as Hausa **?** have different definite articles for noun phrases that have been previously mentioned in contrast to those that are definite by virtue of the situation (e.g., "the podium" at a conference). Definiteness can also be expressed by affixes as in Arabic. **?** shows that Chinese, a language without articles, expresses (in)definiteness through constructions, such as the existential construction for indefinite subjects and the *ba-* construction for definite direct objects. Demonstratives, personal pronouns and possessives (which are found in all languages) are other kinds of definite NPs.

Aside from this variation in the form of (in)definite NPs within and across languages, there is also variability in the semantic, pragmatic, and discourse–related functions (in)definites expresses. We will refer to these as *communicative functions*. The literature on definiteness suggests that it may express communicative functions such as uniqueness, familiarity, identifiability, anaphoricity, specificity, and referentiality (**??????????????**, *inter alia*).

Reductionist approaches to definiteness try to define one or two communicative functions of definiteness. For example, **??** propose that semantic uniqueness is the main communicative function of definite NPs. **?** proposes that the combination of uniqueness and a presupposition of familiarity underlie all definite descriptions. However, possessive definite descriptions (*John's daughter*) and the weak definites (*the son of Queen Juliana of the Netherlands*) are neither unique nor necessarily familiar to the listener before they are spoken.

We take such linguistic observations to suggest that definiteness is not as homogeneous a category as many accounts have assumed. In contrast to the reductionists, we are following an approach to grammaticalization (**?**) in which grammar develops over time in such a way that each grammatical

construction has some prototypical communicative functions, but also has many non-prototypical communicative functions.

This paper describes a classifier that predicts communicative function labels for English noun phrases. The features used by the classifier are lexical, morphological, and syntactic. The contribution of our work is in both the output of the classifier and the model that it uses (features and weights). The classifier outputs communicative function labels (see next section, e.g., whether the entities are old or new to the discourse and to the hearer). Communicative function is important because it is usually preserved in translation even when the grammatical mechanisms for expressing it are different. The communicative function labels also represent the discourse status of entities, making them relevant for entity tracking, knowledge base construction, and information extraction.

The model is a form-meaning mapping, consisting of the syntactic, lexical, and morphological features and weights that correlate with—are predictive of—communicative functions. This in itself is linguistically significant in that it shows the grammatical mechanisms beyond the articles *the* and *a* that are used for expressing definiteness in English. In future work we will build such models for languages that do not have articles such as Hindi, Russian, and Chinese. The form-meaning mapping for these languages can be used for machine translation applications. It has been noted previously that machine translation systems face problems while translating from one language to another when the languages use different grammatical strategies. **??** mention how translating from an article-language to an articleless language is problematic. If the mapping between different forms and the information they encode is known in both the source language and the target language, this information can be leveraged in improving machine translation across these languages.

In §2, we review the annotation scheme we have used to determine the mapping between the form and meaning it encodes with respect to (in)definiteness. In §3, we provide information about the English data we have annotated using this scheme. §4 is an overview of the classification framework we have used. In §5, we describe the measures we have used to evaluate the predictions against the gold standard for the held-out data. In §6, we describe our experiments and discuss the results. In §8, we summarize our findings and briefly discuss the future tasks we are undertaking or envision. [Y T I think that detailed description of definiteness in linguistics in the beginning of this section deserves a section by itself; I'd split the Introduction section into Introduction (shorter, only what we do, motivation, contribution) and then Definiteness.]

2 Annotation scheme

Based on the literature on definiteness, we compiled a list of semantic, pragmatic, and discourse–related features which are relevant for the form of (in)definite NPs in English (and many other languages).[Y T ref to the LREC paper?] These features were put together in a hierarchical structure to ease the annotation effort for the annotators. At each step, the annotators make a decision and thus reduce the number of labels they have to consider before annotating an NP. This approach also increases the consistency in annotations due to the reduced effort on considering the options among which to select the correct label for the NP. The annotation scheme was revised through many iterations after attempts at annotating natural language data from different genres using the scheme. The current scheme is presented in Figure 3.

The first distinction annotators make is whether the NP at hand is anaphoric (discourse old) or non anaphoric (discourse new). We do not annotate for uniqueness, hearer old/new distinction, specificity or genericity for the anaphoric NPs as those distinctions were deemed inconsequential for English anaphoric NPs[1]. [Y T For someone who is not familiar with the scheme, this would be very hard to understand without looking at the definiteness scheme. I think the scheme should be on this page, and description should always point to the place in the scheme. May be we can write all scheme-related terms using the same font as they appear in the scheme?] The category of anaphoric NPs is further subdivided into basic anaphora and extended anaphora. The basic anaphora is used for entities which have been mentioned previously in the discourse. The extended anaphora is inspired by the bridging references **?** had mentioned. These are used for entities which are not

[1] However, in future, we may include these features for anaphoric NPs as well if contrary evidence were found.

(2) I went to *a wedding* last weekend. *The bride* was a friend of mine. She baked *the cake* herself.

Figure 1: [$^{NS}_{S}$ need a caption here]

discourse old or even hearer old, but at the same time, they are not entirely new either. See an example in §2 (borrowed from **?**).

Within the non anaphoric category, the annotators decide whether the NP denotes a unique NP, a nonunique NP, or a generic NP. For the non-generic cases, further decisions are made based on whether the entity is hearer-old or -new, and if it is old, what kind of situation makes it old. For generics also, there are two categories depending on whether they appear with kind-level or individual-level predicates resulting in different properties of these NPs. Besides these, there are a few non-referential categories as well.

Besides the annotation categories, a few other decisions were also taken regarding annotations. For example, corresponding to the basic anaphora cases, the anaphoric links can be established with antecedent NPs in previous sentences as well as with NPs within the same sentence. Also to determine whether an NP gets a SAME_HEAD or a DIFFERENT_HEAD label, we refer to the closest antecedent even if it appears within the same sentence. Regarding annotations of nested NPs, we assume that these NPs are interpreted inside out. This helps determine the cases of BRIDGING_RESTRICTIVEMODIFIER category. If an NP consists of a modifier that restricts its reference enough to make the referent identifiable to the addressees, the embedding NP is annotated with the BRIDGING_RESTRICTIVEMODIFIER label.

Figure 2 is an excerpt from the "Little Red Riding Hood" document demonstrating the above-mentioned decisions regarding annotations and illustrating some of the categories from our annotation scheme.

[$^{NS}_{S}$ we should probably have a name for the scheme—something like "Functions of Definiteness Across Languages (FDAL)"? and a name for the categories (I have been calling them semantic functions).][$^{A}_{B}$ I actually like the name "Functions of Definiteness Across Languages" but am not sure how to incorporate it here.] [$^{NS}_{S}$ suggestion: lead with a discourse excerpt, ideally illustrating a few of the categories, a nested NP, and an anaphoric link across sentence boundaries. I can help format it.]

[$^{NS}_{S}$ make sure to cite related computational stuff such as (**?**)]

Examples from **?**, pp. 6–7:

(1) a. He went to **the bank**. (def.)
Il est allé à **la banque**. (def.)

b. He showed **extreme care**. (unmarked)
Il montra **un soin extrême**. (indef.)

c. I love **artichokes** and asparagus. (unmarked)
J'aime **les artichauts** et les asperges. (def.)

d. His brother became **a soldier**. (indef.)
Son frère est devenu **soldat**. (unmarked)

In the scheme used here, these should receive labels NONUNIQ_HEARER_NEW_SPEC, OTHER_NONREFERENTIAL, GENERIC_INDIVIDUALLEVEL, and PREDICATIVE_EQUATIVE_ROLE, respectively.

3 Data

We used the definiteness corpus **?** which consisted of texts from a few genres, namely TED talks, published news articles, speech (Presidential inauguration speech) and fictional narratives in English. However, currently majority of the corpus consists of the TED talks' annotations (about 72% of the data) with 18% of speech and 10% of news and fiction narratives data. We used a total of 812 sentences (a total of 20655 words), with 2950 NPs (the annotatable units). **?** reports an inter annotator agreement of Cohen's Kappa = 0.94 within the TED genre and 0.91 for combined genres on a previous version of the annotation scheme.

4 Classification framework

To model the relationship between the grammar of definiteness and its semantic functions in a data-driven fashion, we work within the supervised framework of feature-rich discriminative classification, treating the functional categories from §2 as output labels y and various lexical, morphological, and syntactic characteristics of the language as features of the input x. Specifically, we learn a probabilistic log-linear model similar to multiclass logistic regression, but deviating in that logistic regression treats each output label (response) as atomic, whereas we decompose each into *attributes* based on their linguistic definitions, enabling commonalities between related labels to be recognized. Each weight in the model corresponds

Once upon a time there was a dear little girl who was loved by everyone who looked at her, but most of all by her grandmother, and there was nothing that she would not have given to the child.

Once she (SAME_HEAD) gave her (DIFFERENT_HEAD) a little riding hood of red velvet (OTHER_NONREFERENTIAL), which suited her (SAME_HEAD) so well that (NONUNIQ_HEARER_NEW_SPEC) she (SAME_HEAD) would never wear anything else (QUANTIFIED); so she (SAME_HEAD) was always called 'Little Red Riding Hood.' (UNIQ_HEARER_NEW)

Figure 2: An annotated sentence from "Little Red Riding Hood." The previous sentence is shown for context.

- **NONANAPHORA** $[-A,-B]$ **(00)**
  - **UNIQUE** $[+U]$ **(00)**
    - ∗ **UNIQ_HEARER_OLD** $[-G,+O,+S]$ **(00)**
      - · UNIQ_PHYSICAL_COPRESENCE $[+R]$ (00)
      - · UNIQ_LARGER_SITUATION $[+R]$ (00)
      - · UNIQ_PREDICATIVE_IDENTITY $[+P]$ (00)
    - ∗ UNIQ_HEARER_NEW $[-O]$
  - **NONUNIQUE** $[-U]$
    - ∗ **NONUNIQ_HEARER_OLD** $[+O]$
      - · NONUNIQ_PHYSICAL_COPRESENCE $[-G,+R,+S]$
      - · NONUNIQ_LARGER_SITUATION $[-G,+R,+S]$
      - · NONUNIQ_PREDICATIVE_IDENTITY $[+P]$
    - ∗ NONUNIQ_HEARER_NEW_SPEC $[-G,-O,+R,+S]$
    - ∗ NONUNIQ_NONSPEC $[-G,-S]$
  - **GENERIC** $[+G,-R]$
    - ∗ GENERIC_KINDLEVEL
    - ∗ GENERIC_INDIVIDUALLEVEL
- **ANAPHORA** $[+A]$
  - **BASIC** $[+O,-B]$
    - ∗ SAME_HEAD
    - ∗ DIFFERENT_HEAD
  - **EXTENDED** $[+B]$
    - ∗ BRIDGING_NOMINAL $[-G,+R,+S]$
    - ∗ BRIDGING_EVENT $[+R,+S]$
    - ∗ BRIDGING_RESTRICTIVEMODIFIER $[-G,+S]$
    - ∗ BRIDGING_SUBTYPE_INSTANCE $[-G]$
    - ∗ BRIDGING_OTHERCONTEXT $[+O]$
- **MISCELLANEOUS** $[-R]$
  - PLEONASTIC $[-B,-P]$
  - QUANTIFIED
  - PREDICATIVE_EQUATIVE_ROLE $[-B,+P]$
  - PART_OF_NONCOMPOSITIONAL_MWE
  - MEASURE_NONREFERENTIAL
  - OTHER_NONREFERENTIAL

Figure 3: Taxonomy of definiteness functions, with number of occurrences in the training data [$^{NS}_{S}$ TODO]. Internal (non-leaf) labels are in bold; these are not annotated or predicted.[$^{NS}_{S}$ TODO: normalize capitalization] +/− values are shown for ternary attributes Anaphoric, Bridging, Generic, Hearer-Old, Predicative, Referential, Specific, and Unique; these are inherited from supercategories, but otherwise default to 0. Thus, for example, the full attribute specification for UNIQ_PHYSICAL_COPRESENCE is $[-A,-B,-G,+O,0P,+R,+S,+U]$.

to a feature that mediates between *percepts* (characteristics of the input noun phrase) and attributes (characteristics of the label). This is aimed at attaining better predictive accuracy as well as feature weights that better describe the form–function interactions we are interested in recovering.

Our setup is formalized below, where we discuss the mathematical model and linguistically motivated features.

4.1 Model

At test time, we model the probability of semantic label y conditional on a [$^{NS}_{S}$ gold?] noun phrase x as follows:

$$p_{\boldsymbol{\theta}}(y|x) = \log \frac{\exp \boldsymbol{\theta}^\top \mathbf{f}(x,y)}{\sum_{y' \in \mathcal{Y}} \exp \boldsymbol{\theta}^\top \mathbf{f}(x,y')} \quad (1)$$

where $\boldsymbol{\theta} \in \mathbb{R}^d$ is a vector of parameters (feature weights), and $\mathbf{f} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^d$ is the feature function over input–label pairs. The feature function is defined as follows:

$$\mathbf{f}(x,y) = \boldsymbol{\phi}(x) \times \tilde{\boldsymbol{\omega}}(y) \quad (2)$$

where the percept function $\boldsymbol{\phi} : \mathcal{X} \to \mathbb{R}^c$ produces a vector of real-valued characteristics of the input, and the attribute function $\tilde{\boldsymbol{\omega}} : \mathcal{Y} \to \{0,1\}^a$ encodes characteristics of each label. There is a feature for every percept–attribute pairing: so $d = c \cdot a$ and $f_{(i-1)a+j}(x,y) = \phi_i(x)\tilde{\omega}_j(y), 1 \le i \le c, 1 \le j \le a$. The contents of the percept and attribute functions are detailed in §§4.2 and 4.3.

For prediction, having learned weights $\hat{\boldsymbol{\theta}}$ we choose the y that maximizes this probability:

$$\hat{y} \leftarrow \arg\max_{y' \in \mathcal{Y}} p_{\hat{\boldsymbol{\theta}}}(y|x) \quad (3)$$

Training optimizes $\hat{\boldsymbol{\theta}}$ so as to maximize a convex L_1-regularized learning objective over the training data $\mathcal{D}$:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}, \mathcal{D}) \quad (4)$$

$$L(\boldsymbol{\theta}, \mathcal{D}) = -\lambda ||\boldsymbol{\theta}||_1 + \sum_{\langle x,y \rangle \in \mathcal{D}} \log \frac{\exp \boldsymbol{\theta}^\top \mathbf{f}(x,y)}{\sum_{y' \in \mathcal{Y}} \exp \left(\boldsymbol{\theta}^\top \mathbf{f}(x,y') \right)} \quad (5)$$

With $\tilde{\boldsymbol{\omega}}(y)$ = *the identity of the label*, this reduces to standard logistic regression.

4.2 Percepts

The characteristics of the input that are incorporated in the model, which we call *percepts* to distinguish them from model features linking inputs to outputs,[2] are intended to capture the aspects of English morphosyntax that may be relevant to the semantic and pragmatic functions of definiteness.

After preprocessing the text with a dependency parser and coreference resolver, we extract the several kinds of percepts for each noun phrase (NP).

4.2.1 Basic

Words of interest. These are the *head* within the NP, all of its *dependents*, and its *governor* (external to the NP). We are also interested in the *attached verb*, which is the first verb one encounters when traversing the dependency path upward from the head. For each of these words, we have separate percepts capturing: the token, the part-of-speech (POS) tag, the lemma, the dependency relation, and (for the head only) a binary indicator of plurality (determined from the POS tag). As there may be multiple dependents, we have additional features specific to the first and the last one. Moreover, to better capture tense, aspect and modality, we collect the attached verb's *auxiliaries*. We also make note of *neg* if it is attached to the verb.

Structural. These are: the *path length* from the head up to the root, and to the attached verb. We also have percepts for the number of dependents, and the number of dependency relations that link non-neighbors. Integer values were binarized with thresholding.

Positional. The *token length* of the NP; the NP's *location* in the sentence (first or second half); the *attached verb's position* relative to the head (left or right). 12 additional percept templates record the POS and lemma of the left and right neighbors of the head, governor, and attached verb.

4.2.2 Contextual NPs

When extracting features for a given NP (call it the "target"), we also consider NPs in the following relationship with the target NP: its *immediate parent*, which is the smallest NP whose span fully subsumes that of the target; the *immediate child*, which is the largest NP subsumed within the target; the *immediate precedent* and *immediate successor* within the sentence; and the *nearest preceding coreferent mention*.

For each of these related NPs, we include all of their basic percepts conjoined with the nature of the relation to the target.

4.3 Attributes

As noted above, though labels are organized into a tree hierarchy, there are actually several dimensions of commonality that suggest different groupings. These attributes are encoded as ternary characteristics; for each label (including internal labels), every one of the 8 attributes is assigned a value of +, −, or 0 (refer to fig. 3). In order to capture these similarities in the model's features, we define the attribute vector function $\boldsymbol{\omega}(y)$ =

$$[y, A(y), B(y), G(y), O(y), P(y), R(y), S(y), U(y)]^\top$$

where $A : \mathcal{Y} \rightarrow \{+, -, 0\}$ returns the value for Anaphoric, $B(y)$ for Bridging, etc. The identity of the label is also included in the vector so that different labels are always recognized as different by the attribute function. The categorical components of this vector are then binarized to form $\tilde{\boldsymbol{\omega}}(y)$; however, instead of a binary component that fires for the 0 value of each ternary attribute, there is a component that fires for *any* value of the attribute—a sort of bias term. The weights assigned to features incorporating + or − attribute values, then, are easily interpreted as deviations relative to the bias.

5 Evaluation

The following measures will be used to evaluate our predictor against the gold standard for the held-out evaluation (dev or test) set $\mathcal{E}$:

- **Exact match:** This accuracy measure gives credit only where the predicted and gold labels are identical.
- **By leaf label:** We also compute precision and recall of each leaf label to determine which categories are reliably predicted.
- **Soft match:** This accuracy measure gives partial credit where the predicted and gold labels are related. It is computed as the proportion of attributes whose (categorical) values match: $|\boldsymbol{\omega}(y) \cap \boldsymbol{\omega}(y')|/9$.
- **Perplexity:** This determines how "surprised" our model is by the gold labels in the test set;

[2] See above.

the greater the probability mass assigned to the true labels, the higher the score. It is computed as $2^{\left(\sum_{\langle x,y\rangle \in \mathcal{E}} \log_2 p_{\hat{\theta}}(y|x)\right)/|\mathcal{E}|}$.

6 Experiments

6.1 Experimental Setup

The annotated corpus of **?** (§3) contains 16 documents in 3 genres: 12 prepared speeches (mostly TED talks), 2 newspaper articles, and 2 fictional narratives. We arbitrarily choose some documents to hold out from each genre; the resulting test set consists of 2 TED talks ("Alisa_News", "RobertHammond_park"), 1 newspaper article ("crime1_iPad_E"), and 1 narrative ("Little Red Riding Hood"). The test set then contains 3,558 tokens (110 sentences), in which there are 492 annotated NPs; while the training set contains 2,458 NPs among 17,097 tokens (702 sentences). Gold NP boundaries are assumed throughout our experiments.

We use an in-house implementation of supervised learning with L_1-regularized AdaGrad (**?**). Hyperparameters are tuned on a dev set formed by holding out every tenth instance from the training set (test set experiments use the full training set).[$^{\text{NS}}_{\text{S}}$ early stopping? what exactly is tuned? optimize soft match acc?] Automatic dependency parses and coreference information were obtained with the parser and coreference resolution system in Stanford CoreNLP v. 3.3.0 (**??**) for use in features (§4.2).

6.2 Results

[$^{\text{NS}}_{\text{S}}$ English: ±cost function, ±non-identity attributes, ±predicting intermediate labels]

[$^{\text{NS}}_{\text{S}}$ maybe: which attribute groupings produce the best classifier, if we want to force a hierarchy]

[$^{\text{NS}}_{\text{S}}$ feature/attribute ablations]

[$^{\text{NS}}_{\text{S}}$ Hindi?]

7 Related Work

[$^{\text{NS}}_{\text{S}}$ computational approaches to things related to definiteness, e.g. in MT. also would be good to mention Bresnan's work on predicting syntactic alternations with logistic regression (here we want to predict the hidden information so that the classifier is useful for applications!).]

8 Conclusion

Condition	$\lvert\boldsymbol{\theta}\rvert$	Exact Match Accuracy	Soft Match Accuracy	Perplexity
Majority baseline	—			
Log-linear classifier, no grouping by attributes				
Full log-linear classifier				

Table 1: Classifier versus baselines.