Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Pfaffenwaldring 5B
D-70569 Stuttgart

Bachelor Thesis

Automatic Classification of Abstractness in English Rigid Nouns

Alberto Saponaro

Studiengang:	B.Sc. Maschinelle Sprachverarbeitung
Prüferin:	Prof. Dr. Sabine Schulte im Walde
Betreuerin:	Prof. Dr. Sabine Schulte im Walde
Beginn der Arbeit:	01.05.2023
Ende der Arbeit:	01.11.2023

Erklärung (Statement of Authorship)

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und dabei keine andere als die angegebene Literatur verwendet habe. Alle Zitate und sinngemäßen Entlehnungen sind als solche unter genauer Angabe der Quelle gekennzeichnet. Die eingereichte Arbeit ist weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen. Sie ist weder vollständig noch in Teilen bereits veröffentlicht. Die beigefügte elektronische Version stimmt mit dem Druckexemplar überein.[1]

(Alberto Saponaro)

[1] Non-binding translation for convenience: This thesis is the result of my own independent work, and any material from work of others which is used either verbatim or indirectly in the text is credited to the author including details about the exact source in the text. This work has not been part of any other previous examination, neither completely nor in parts. It has neither completely nor partially been published before. The submitted electronic version is identical to this print version.

Abstract

The main difference between (i) Mass-Count Languages (such as English) and (ii) Classifiers Languages (such as Chinese) is that (i) encode the information about nouns' countability in their grammar and (ii) employ a classification system of classifiers to distinguish between individuals or substance. If the mass-count distinction is a characteristic of mass-count language, the substance-individuals denotation seems to be a concept universally available for all humans. Another concept that appears to be universally accessible and linked to the countability status of English nouns is the notion of abstractness. Then, mass nouns usually refer to an abstract object, and this is confirmed from the distribution of abstractness in the dataset.

This thesis' objective is to provide a model for the classification of rigid nouns (count or mass only) that is capable to generalize on the degree of abstractness. Additionally, it tests if a model trained with the same set of features is capable of rating the abstractness of those nouns. To accomplish these tasks, several sets of features are being identified based on syntactic and semantic properties of nouns that describe the mass-count distinction.

The results indicate that the first model M_1, a mass-count classifier that predicts the countability class of a rigid noun, provides reliable predictions and can generalize on the degree of abstractness of the targets. The second model M_2, an abstractness rate predictor that assigns an abstractness rate from 1 to 5 to a rigid noun, is incapable of providing reliable ratings and cannot generalize on the countability status of the targets. A third model M_3, an abstract-concrete (binary) classifier that predicts the abstractness class of a rigid noun, provides reliable predictions and can generalize on the countability status of the targets.

Given that those results concerns rigid nouns only, further research can be conducted by examining the abstractness of elastic nouns. However, there is the need of an annotation that rates abstractness of nouns senses.

Kurzfassung

Der Hauptunterschied zwischen (i) Mass-Count Sprachen (wie Englisch) und (ii) Klassifizierer Sprachen (wie Chinesisch) besteht darin, dass (i) die Information über die Zählbarkeit der Nomen in ihrer Grammatik codieren und (ii) ein System von Klassifikatoren verwenden, um zwischen Individuen und Substanz zu unterscheiden. Wenn der Mass-Count Distinktion eine Eigenschaft von Mass-Count Sprachen ist, die Individuen-Substanz Denotation scheint ein Konzept zu sein, das allen Menschen universell zugänglich ist. Ein weiteres Konzept, der universell zugänglich zu sein scheint und mit dem Zählbarkeitsstatus englischer Nomen verbunden ist, ist das Konzept der Abstraktheit. Massennomen beziehen sich normalerweise auf abstrakte Objekte, und dies wird durch die Verteilung der Abstraktheit in dem Datensatz bestätigt.

Das Ziel der Thesis ist, ein Modell für die Klassifizierung von starren Nomen (nur Masse oder Zählbar) bereitzustellen, das in der Lage ist, auf den Abstraktheitsgrad zu generalisieren. Ebenso wird getestet, ob ein Model, das mit demselben Featuressatz trainiert wurde, in der Lage ist, den Abstraktheitsgrad dieser Nomen zu bewerten. Um diese Aufgabe zu erfüllen, wurden mehrere Featuressätze identifiziert, die auf syntaktischen und semantischen Eigenschaften von Nomen basieren, die die Mass-Count Distinktion beschreiben.

Die Ergebnisse zeigen, dass das erste Modell M_1, ein Mass-Count Klassifikator, der die Zählbarkreisklassen eines starren Nomens vorhergesagt, zuverlässige Vorhersagen liefert und auf den Abstraktheitsgrad der Ziele generalisieren kann. Das zweite Modell M_2, ein Abstraktheitsbewertung Prädiktor, der einem starren Nomen eine Abstraktheitsbewertung von 1 bis 5 zuweist, ist nicht in der Lage, zuverlässige Bewertungen zu liefern und kann nicht auf den Vorhersagt, liefert zuverlässige und kann auf den Zählbarkeitsstatus der Ziele generalisieren. Ein drittes Modell M_3, ein Abstrakt-Konkret Klassifikator, der die Abstraktheitsklasse eines starren Nomens vorhersagt, liefert zuverlässige Vorhersagen und kann auf den Zählbarkeitsstatus der Ziele generalisieren.

Da diese Ergebnisse nur starre Nomen betreffen, können weitere Untersuchungen durchgeführt werden, indem die Abstraktheit elastischer Nomen untersucht wird. Allerdings besteht die Notwendigkeit einer Annotation, die den Abstraktheitsgrad von Nomen-Sinnen bewertet.

Contents

List of Figures

List of Tables

1 Introduction

There is more than one way to express countability in languages (Chierchia, 2010). Languages, such as English, that employ a grammaticized mass-count distinction are called mass-count languages. Nouns in such languages typically can be considered *mass* and/or *count*. Taking a look at those nouns, one can observe that mass nouns define (A) *stuff* that can be amounted and count nouns define (B) *things* that can be counted (Ware, 1979). Concepts of countability are not only understood by native speakers of mass-count languages, but are available to all humans. A property of nominal roots that is available across languages is the substance-individuals denotation, and this property seems to be related to the mass-count distinction. Then, mass nouns usually denote *substance* (*stuff*) and count nouns usually denote *individuals* (*things*).

Example:

(A) affect$_{\texttt{+mass}}$: denotes substance and can be amounted

(B) actor$_{\texttt{+count}}$: denotes individuals and can be counted

Another concept that is universally available across humans like the substance-individuals denotation is the notion of abstractness. In his paper, Zamparelli (2020) points out that abstract nouns generally do not refer to a concrete *object*. Furthermore, concrete *objects* should impinge more on the five senses (smelling, tasting, touching, hearing, seeing). Based on the criterions used to describe *concreteness* in Brysbaert et al. (2013) to collect ratings for thousands of words, one should expect mass nouns to be more abstract than count nouns (H1).

Example:

(C) affect$_{\texttt{+mass}}$: has a degree of abstractness of 4.07 out of 5.

(D) actor$_{\texttt{+count}}$: has a degree of abstractness of 1.43 out of 5.

(H1) Mass nouns, as they usually denote substance, tend to be more abstract than count nouns.

This thesis attempts to identify a set of properties/features that contains information about the countability status[1] of English nouns (Q1) and at the same time can provide some hints about their degree of abstractness[2] (Q2).

(Q1) What features should be extracted from a corpus to better describe the distinction between mass and count in English nouns?

(Q2) Can the same features extracted from a corpus to describe the mass-count distinction in English nouns be suitable to describe the degree of abstractness of those nouns? And with which resolution (binary or multi-class)?

To determine the countability status of a noun, this thesis utilizes the BECL 2.1 annotation (Kiss et al., 2016), which provides the countability status for nouns' senses (polysemy). The degree of abstractness is provided by the Brysbaert's Concreteness Ratings (Brysbaert et al., 2013), which provides the mean of the ratings for each noun. Because the two gold standards differ in their granularity, this thesis examines

[1]If a noun is classified as being mass or count.

[2]How abstract is a noun on a scale of 1 to 5.

rigid[3] nouns only. Then, *elastic* nouns can shift in countability (Zamparelli, 2020), making their countability status *undefined* on the *word* level.

Examples:

(E) Walter drinks wine$_{\texttt{+mass}}$.
wine$_{\texttt{+mass}}$: denotes substance and refers to the *fluid*.

(F) Walter bought two wines$_{\texttt{+count}}$.
wine$_{\texttt{+count}}$: denotes individuals and refers to a *variety* of wine.

An alternative would be assigning the same abstractness degree to all senses of a noun, which would lead to an "a priori" falsification of (H1). Additionally, the study conducted in Reijnierse et al. (2019) brings some evidence that the degree of concreteness of a noun could vary based on its meaning. Instead, for *rigid* nouns, the BECL annotation and the Brysbaert Concreteness Ratings can be combined to establish a gold standard (like in (C) and (D)) to test (H1) and evaluate the models.

With a dataset providing the *abstractness degrees of rigid nouns*, the thesis can attempt to find out which properties/features are responsible for the mass-count distinction in English (Q1) by training a model M to classify mass and count nouns (H2). Then, to find out how well the model M can describe the abstractness of the target nouns (Q2), the thesis makes it generalize on the abstractness of those (H2). To better research on the features (originally extracted to classify mass and count nouns) and their capability to describe the abstractness of the rigid nouns (Q2), a model M' is being trained to test if it can reliably make predictions on the abstractness of those nouns (H3). At the end, similar to (H1), it is interesting to observe if the model M' can generalize on the countability status of its targets (H4).

[3]Rigid nouns are either mass or count only. In the BECL annotation, they are all grouped under the classes 538 (mass only) and 235 (count only).

(H2) A model M, trained to classify mass and count nouns, can generalize on the degree of abstractness of those nouns.

(H3) A model M', trained with the same features as M, can reliably rate the abstractness of a noun.

(H4) The model M', can generalize on the countability status of the nouns.

The thesis is structured as follows:

Chapter 2: The Mass-Count Distinction

This chapter provides a brief overview of the literature on the mass-count distinction in English and cross-language.

Section 2.1 discusses the differences between mass-count and classifiers languages.

Section 2.2 discusses the mass-count distinction in English.

Section 2.3 discusses the abstractness of nouns.

Chapter 3: Materials and Methods

This chapter discusses the materials and methods used to test the hypotheses.

Section 3.1 explores the datasets and establishes a gold standard for the classification tasks.

Section 3.2 illustrates how the features used for the nouns' representations were extracted from the ENCOW corpus.

Section 3.3 describes the methods used to implement the models.

Chapter 4: Results and Discussion

This chapter discusses the results and answers the research questions.

Section 4.1 illustrates the results achieved by the models.

Section 4.2 discusses the overall results of the thesis and future work.

2 The Mass-Count Distinction

There are at least two ways to express basic concepts of countability across languages: (i) with a grammaticized mass-count distinction, and/or (ii) with a classifiers' system. The geographical distribution of those languages reveals that languages (i) are mostly Indo-European and languages (ii) are mostly Asian (Chierchia, 2010). Chierchia (2010) identifies a third category of languages (iii) which lacks both (i) and (ii), those languages are mostly Amerindian. This thesis focuses on languages (i), also called mass-count languages. Since the mass-count distinction in English is the subject of this investigation, it is appropriate to provide a general definition for mass and count nouns.

Definition: mass and count nouns

Count nouns identify units that can be counted. **Mass** nouns name entities that come in *mass* form and cannot be separated into countable units (Ghomeshi and Massam, 2012).

In mass-count languages, **count** nouns usually denote **individuals**, and **mass** nouns usually denote **substance**.

To gain a better understanding of the semantic underlining the mass-count distinction in mass-count languages, the formers can be compared to classifiers languages.

2.1 Mass-Count and Classifiers Languages

The literature suggests that the mass-count distinction is not universally grammaticized, and there are several ways in which languages can express countability. Wiltschko (2012) investigates on Blackfoot and Halkomelem observing that they do not classify their nouns as being mass or count. However, native speakers of those two languages make a distinction between individuals and substance. The main difference between these two languages and a mass-count language such as English is that their grammar does not take care of the individuals-substance denotation. Moreover, if native speakers of mass-count languages are able to distinguish individuals from substance, the information about individuals-substance denotation should be encoded into the mental lexicon of all humans. Then, mass nouns usually denote substance and count noun individuals. In Chinese, a classifiers language, it can be observed that all nouns are being categorized as being *mass* and later only be recategorized by the classifier system as being *non-mass*. Instead, mass-count languages such as English categorize individual nouns as being mass or count, without needing a classifier system to take on this task (Ghomeshi and Massam, 2012).

An example of the lack of grammaticized mass-count grammar in Mandarin in comparison to English is the co-occurrence of *number words* (*one*, *two*, *three*, etc.) with nouns. Then in English, nouns can co-occur directly with *number words*, with the requirement to be (1) count nouns and (2) those count nouns need to be pluralized (with the exception being the number *one*). In contrast to English, in Mandarin, a classifiers language, *number words* cannot co-occur directly with most of the nouns (an exception is *ren* (person/people)), but require classifiers. Those classifiers are words akin to English *measure words* (*piece of*, *grain of*, etc.) (Bale and Barner, 2012).

Example:

i. liang *li* mi / !liang mi
two CL rice / two rice
'tow grains of rice'

ii. liang *ge* hazi / !liang hazi
two CL rice children / two children
'two children'

*examples from Bale and Barner (2012)

To better understand how the mass-count distinction operates on nouns, it is necessary to investigate further the source of language variation between mass-count and classifiers languages.

2.1.1 The Source of Language Variation

In languages, the variation on how countability is expressed occurs not only between mass-count and classifiers languages, but also between languages of the same category. An example of this phenomenon is the plural marking in bare plurals. In English, (A) mass plural appear bare, but this behavior is not generally allowed in (B) romance languages (Ghomeshi and Massam, 2012). Therefore, it is important to identify the source of language variation.

Example:

(A) English (non-romance)

i. The bananas are tasty.

ii. Bananas are tasty.

(B) Italian (romance)

i. Le banane sono buone.

ii. ! Banane sono buone.

Wiltschko (2012) identifies two properties of language that should be responsible for the language variation: (1) *ontological* properties and (2) *categorical* properties.

Ontological Properties: individuals-substance denotation

These properties are available universally across languages and describe the things in the world that nominal roots name. These properties are not categorical because there is some ambiguity about nouns denoting substance or individuals.

Categorical Properties: mass-count distinction

These properties are responsible for the distributional differences that distinguish mass nouns from count nouns. The mass-count distinction appears to be categorical, in that there are a number of morphosyntactic diagnostics that divide nominal phrases to be mass or count.

For Bale and Barner (2012) the primary distinction between mass-count and classifiers languages pertains to the matter in which the mass-count distinction is encoded in the syntax of the language. The mass-count syntax does not simply reflect the ontological properties, but has others subtle semantic implications. Wiltschko (2012) argues that categorical and ontological properties are not linked to each other. Not only should not be possible to infer categorical properties from ontological ones, but in mass-count languages nouns should follow the grammar blindly, ignoring the ontological denotation. Bale and Barner (2012) observe that in English count syntax does not only signal individuation, but it could trigger it grammatically (C), and its absence (D) could result in a substance-like interpretation.

Example:

(C) i. Mary bought wine$_{\texttt{+mass}}$. (substance)

ii. Mary bought two wines$_{\texttt{+count}}$. (individuals)

(D) i. Mary has more bananas$_{\texttt{+count}}$ than Jane does. (individuals)

ii. Mary likes banana$_{\texttt{+mass}}$ more than Jane does. (substance)

As an example of nouns that follow grammar blindly, Wiltschko (2012) utilize *object-mass*[4] nouns. Bale and Barner (2012) found a class of nouns in Mandarin with similar properties.

2.1.2 English's Object-Mass and Mandarin's Bare Nouns

Bale and Barner (2012) individuate similarities in English's object-mass nouns like *furniture* and Mandarin's bare nouns such as *pingguo* (*apples*). Those nouns (i) can denote individual lacking count or classifier syntax, (ii) can be used semantically inert classifiers or measures words, and (iii) are underspecified for number, and those can refer to either groups or individuals. They came up with two conclusions about the semantics of those words. First, they have all atomic minimal parts in their denotations. And second, they both do not only contain atomic parts, but

[4]Nouns such as *furniture*, *information* and *jewerly* that reference to an amount of atomic individuals.

also contain all the groups that can be formed from those minimal parts (Bale and Barner, 2012).

Example:

a. Roberto bought *more* furniture from Italy than from Sweden.

b. !Roberto bought *more* furnitures from Italy than from Sweden.

c. Roberto bought two *pieces of* furniture. (individuals)

d. Roberto bought forniture. (group or individual?)

2.2 The Mass-Count distinction in English

The categorical properties identified in Wiltschko (2012) seem to be those responsible for the mass-count distinction. Pelletier (2012) describes the same properties in English nouns as being syntactic conditions for `+mass` and `+count`.

Syntactic conditions for +mass:

(1) Mass nouns, but not count nouns, do not have plural forms and thus all verb agreement is singular.

Example:

i. Alberto studies biotechnology$_{\texttt{+sg +mass}}$

ii. !Alberto studies biotechnologies$_{\texttt{!+pl +mass}}$

iii. Biotechnology$_{\texttt{+sg +mass}}$ *is*$_{\texttt{+sg}}$ being studied in universities.

iv. !Biotechnology$_{\texttt{+sg +mass}}$ *are*$_{\texttt{!+pl}}$ being studied in universities.

(2) Mass nouns, but not singular count nouns, can occur with measure phrases like *liters of*, *amount of*.

Example:

i. Jane ordered *a ton of* merchandise$_{\texttt{+sg +mass}}$ from an online-shop.

ii. !Jane ordered *a ton of* sweatshirt$_{\texttt{!+sg +count}}$ from an online-shop.

iii. Jane ordered *a ton of* sweatshirts$_{\texttt{+pl +count}}$ from an online-shop.

(3) Mass nouns, but not count nouns, employ the quantifiers *much*, *little*.

Example:

i. Bad work comes from *little* thinking$_{\texttt{+sg +mass}}$.

ii. !Jane ordered *much* sweatshirts$_{\texttt{+sg !+count}}$.

(4) Mass nouns, but not singular count nouns, employ the unstressed *some* and the quantifier *most*.

Example:

i. Jane bought *most* of her merchandise$_{\texttt{+sg +mass}}$ from an online-shop.

ii. !Jane ordered *most* of her sweatshirt$_{\texttt{!+sg +count}}$ from an online-shop.

iii. Jane ordered *most* of her sweatshirts$_{\texttt{+pl +count}}$ from an online-shop.

Syntactic conditions for +count:

(5) Count nouns, but not mass nouns, have plural forms and those can agree with plural verbs.

> **Example:**
>
> i. Linkin Park released a total of seven albums$_{\texttt{+pl +count}}$.
>
> ii. Seven albums$_{\texttt{+pl +count}}$ *have*$_{\texttt{+pl}}$ been released by Linkin Park.
>
> iii. !Linkin Park sell merchiandise<u>s</u>$_{\texttt{! +pl +mass}}$ on their online-store.

(6) Count nouns, but not mass nouns, can occur with *numerals* and *counting phrases*.

> **Example:**
>
> i. Jane bought *two* sweatshirts$_{\texttt{+count}}$.
>
> ii. !Jain bought *two* merchandise$_{\texttt{! +mass}}$.

(7) Singular count nouns, but not mass nouns, employ the quantifiers *each*, *every*, (stressed quantifier) *some*, and definite *a(n)*.

> **Example:**
>
> i. Jane bought *every* sweatshirt$_{\texttt{+count}}$.
>
> ii. !Jain bought *every* merchandise$_{\texttt{! +mass}}$.

(8) Plural count nouns, but not mass nouns, employ the quantifiers few, several, many.

> **Example:**
>
> i. Jane bought *several* sweatshirts$_{\texttt{+count}}$.
>
> ii. !Jain bought *several* merchandise$_{\texttt{! +mass}}$.

Pelletier (2012) also identifies a number of semantic features that should hold across languages. These features resemble the ontological properties described in Wiltschko (2012). They describe the relationship between *mass* and *substance* and the relationship between *count* and *individuals*.

Semantic features of +mass: Mass nouns designate **staff** ($\approx$ substance).

(1) *Mass* is divisive in its reference.

> **Example:**
> i. If merchandise$_{\texttt{+mass}}$ is divided in half, it splits into two groups.
>
> ii. If a sweatshirt$_{\texttt{+count}}$ is cut in half, it is no longer a whole object.

(2) *Mass* is cumulative in its reference.

> **Example:**
> i. merchandise$_{\texttt{+mass}}$ + merchandise$_{\texttt{+mass}}$ = merchandise$_{\texttt{+mass}}$
>
> ii. sweatshirt$_{\texttt{+count}}$ + sweatshirt$_{\texttt{+count}}$ $\neq$ sweatshirt$_{\texttt{+count}}$
> = 2 * sweatshirt$_{\texttt{+count}}$

(3) *Mass* cannot be counted.

> **Example:**
> i. Maria bought *some* merchandise$_{\texttt{+mass}}$.
>
> ii. !Maria bought *two* merchandise$_{\texttt{+mass}}$.

(4) *Mass* can be measured.

> **Example:**
> i. Maria bought *a ton of* merchandise$_{\texttt{+mass}}$.
>
> ii. !Maria bought *a ton of* sweatshirt$_{\texttt{!+sg +count}}$.

Semantic features of +count: Count nouns designate a set of (countable) **things**($\approx$ individuals).

(5) *Counts* are a unit as a whole.

> **Example:**
>
> i. If a sweatshirt$_{\texttt{+count}}$ is cut in half, it is no longer a whole object.
>
> ii. If a sweatshirt$_{\texttt{+count}}$ is copied, two whole objects are obtained.

(6) (Singular) *counts* are not a part in themselves.

> **Example:**
>
> i. A sweatshirt$_{\texttt{+sg +count}}$ is not made of sweatshirt$_{\texttt{+sg +count}}$, but is made of other materials/particles.
>
> ii. Merchandise$_{\texttt{+mass}}$ is made of Merchandise$_{\texttt{+mass}}$.

(7) *Counts* are individuated items that can be counted.

> **Example:**
>
> i. Maria bought *ten* sweatshirts$_{\texttt{+count}}$.
>
> ii. !Maria bought *ten* merchandise$_{\texttt{!+mass}}$.

(8) (Singular) *counts* are not measurable.

> **Example:**
>
> i. !Maria bought *a ton of* sweatshirt$_{\texttt{!+sg +count}}$.
>
> ii. Maria bought *a ton of* merchandise$_{\texttt{+mass}}$.

"Appropriateness" of determiners and quantifiers. Nouns, in their various occurrences, do not always appear with their distinguishing *quantifiers* or *determiners* (Ware, 1979). Then, the syntactic conditions for `+mass` and `+count` are not always mandatory. For example, count nouns can be pluralized (syntactic condition (1)), but they do not always appear in plural form; mass nouns can employ quantifiers like *little* and *much* (syntactic condition (2)), but those are not mandatory. Considering the fact that those conditions are not mandatory, when it comes to *quantifiers* and *determiners* Ware (1979) write about the *appropriateness* of those with mass and/or count nouns.

2.3 The Elasticity of Nouns

The meaning of a noun often depends on the context in which it is being utilized (polysemy). In Zamparelli (2020) is described how the countability of nouns is often *elastic* and a *shift* in countability causes a *shift* in meaning. Meaning that on the *word level,* the countability status of a word can be potentially ambiguous. Then, a word can have multiple *senses*(meanings) and those senses can be either *count* or *mass senses.*

Example:

i. Wine$_{\texttt{+mass}}$ is a beverage made from fermented grapes. (fluid/beverage)

ii. Roberto bought two Tuscan wines$_{\texttt{+count}}$. (variety/bottles)

A dataset that lists the countability status of noun-senses is the BECL (Kiss et al., 2016) annotation. In this annotation, nouns that present a shift in countability are called *elastic nouns*. However, there is another category of nouns that do not present any shift in countability across *senses*(meanings), and those are called *rigid nouns.*

Example:

i. John purchases *plenty of* Star Wars' merchandise$_{\texttt{+mass}}$.

ii. !John purchases *two* Star Wars' merchandise$\underline{\text{s}}_{\texttt{!+count}}$. (rigid `+mass`)

iii. John purchases *a* Star Wars' sweatshirt$_{\texttt{+count}}$.

iv. !John purchases *plenty of* Star Wars' sweatshirt$_{\texttt{!+mass}}$. (rigid `+count`)

This thesis focuses on *rigid nouns*, which have a fixed countability status, even if they can be polysemous[5]. The fact that *rigid nouns* are either *mass* or *count only* helps to solve the issues with granularity that were mentioned in chapter 1, and allows for the creation of a subset that merges information about *countability* and *abstractness* of *rigid nouns*.

2.4 The Abstractness of Nouns

Abstractness is a really broad concept, so this thesis only focuses on the essentials aspects that are relevant to the mass-count distinction. To define *abstractness*, the thesis utilizes the definition of *concreteness* from Brysbaert et al. (2013) and derives the reverse scale. For example, if a patient rated the word *worker* with a *concreteness rate* of 5 out of 5, then the corresponding *abstractness* rate for *worker* will be 1 out of 5. The following definition of *concreteness* was provided in Brysbaert et al. (2013) to help participants of the study rating words:

[5] More than one meaning.

Definition: Concreteness (Brysbaert et al., 2013)

Concrete Words refer to things that exist in reality. They can be experienced through the five senses (smelling, tasting, touching, hearing, seeing) and the action someone does. The easiest way to explain those words is by pointing to them or by demonstrating them.

Abstract words refer to things that cannot be experienced in reality, and their meaning depends on language. The easiest way to explain them is by using other words.

Zamparelli (2020) points out that corpus-based research suggests that the majority of mass nouns are abstract. Then, mass nouns usually denote *substance* and count nouns *individuals*. With the definition of *concreteness* provided in Brysbaert et al. (2013) it can be said that *individuals* usually point to concrete *objects* that exist in reality and can be experienced by the human perception and *substance* usually refers to abstract *objects* that cannot be experienced through the human perception.

Example:

i. $worker_{\texttt{+count}}$ (concrete, abstractness degree of 1.41) points to a person who is working and exists in reality.

ii. $workmanship_{\texttt{+mass}}$ (concrete, abstractness degree of 1.41) refers to the quality of the work done by a worker, does not exist in reality and is negotiable (subjective).

3 Materials and Methods

This section discusses the materials and methods used in the thesis to research on the features that better describe the mass-count distinction of English nouns (Q1), and if the same features are suitable to describe the degree of abstractness of those (Q2). The process starts by looking at the datasets to later establish a gold standard for the *abstractness degrees of rigid nouns*. Then on the gold standard it can be tested if mass nouns tend to more abstract than mass nouns (H1). After the data exploration, the knowledge provided by the literature in chapter 2 is utilized to extract features from a corpus that can be used as word representation for the noun that this thesis utilizes targets. Finally, the models used for the classification tasks are presented.

3.1 Dataset Exploration and The Gold Standard

In this section, two tasks are carried out: (1) explore the datasets to gain a better understanding of the data, and (2) combine the BECL annotation and the Brysbaert norm into a single gold standard to evaluate the classifiers.

3.1.1 The Bochum English Countability Lexicon (BECL 2.1)

In its 2.1 version, the *Bochum English Countability Lexicon* (BECL) large-scale annotation project lists the countability status for 11869 *noun-senses* (polysemy). Figure 3.1 is a tree view of the countability classes present in the BECL 2.1 annotation (Kiss et al., 2016). *Noun-Senses* are classified into four *major classes* and 18 *(sub)classes* (Figure 3.1) and the majority of *noun-senses* are classified under *regular count* (Table 3.2). *Nouns-senses* that are *both mass and count* or *neither mass nor count* are only a small portion of the dataset (Table 3.2).

id	sense	lemma	wordnet _senseindex _number	...	wordnet _total _senses	...	occurrences _in_oanc_total	...	class	major_class	...
25085	2	aa	2	...	3	...	306	...	523	neither _mass_count	...
25085	3	aa	3	...	3	...	306	...	235	regular_count	...
40178	1	abbreviation	1	...	2	...	140	...	235	regular_count	...
20030	1	aberrancy	1	...	1	...	14	...	235	regular_count	...
24831	1	aberration	1	...	3	...	30	...	235	regular_count	...
...	...	...	...	...	...	...	...	...	...	...	...
25887	1	zoo	1	...	1	...	99	...	235	regular_count	...
413	1	zygote	1	...	1	...	12	...	235	regular_count	...

Table 3.1: Snippet of The BECL 2.1 Annotation Dataset.

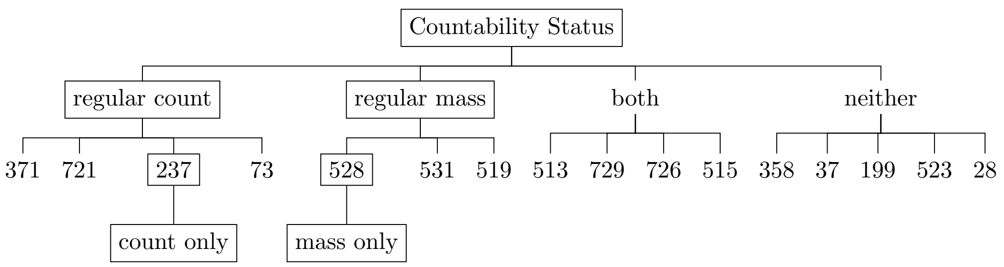


Figure 3.1: BECL 2.1 Countability Classes Tree

	all	regular mass	regular count	both mass and count	neither mass nor count
count	11869	2425	8432	697	315
%	100	20.43	71.04	5.87	2.65
mean	3.34	2.94	3.49	2.73	3.57
std	2.52	2.02	2.69	1.75	2.11
min	1	1	1	1	1
25%	2	1	2	1	2
50%	3	1	3	2	3
75%	4	4	4	4	5
max	33	17	33	12	12

Table 3.2: Descriptive Statistics of Noun-Senses in BECL 2.1

The classes examined in this thesis are those under the major classes *regular count* and *regular mass*. Figure 3.2 is a count plot that describes how often the *major classes* (*regula mass* and *regular count*) have been seen in the BECL 2.1 annotation. The *major classes* are placed on the y-axis and their occurrences on the x-axis. The graph shows the baseline being unbalanced, with *regular count senses* occurring almost four times more frequently than *mass senses* (Table 3.2). This is something to keep in mind, as the baseline should be balanced before the model is trained.

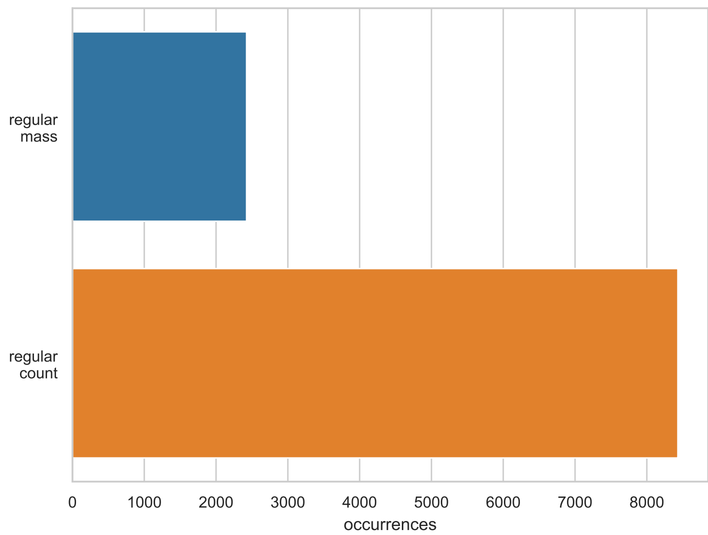


Figure 3.2: Distribution of *Regular* Nouns in BECL 2.1

The BECL 2.1 annotation also provides information on the countability status of *noun-senses*, the total number of WordNet *noun-senses*, and the occurrences of nouns (not *noun-senses*) in *The Open Americans National Corpus* (OANC). Figure 3.3a shows the degree of polysemy for nouns in the OANC corpus. The WordNet total number of senses that a noun could possess are placed on the x-axis (degree of polysemy). Most of the outliers were taken out of the box plot for readability reasons. The median number of senses for a noun in the OANC is 3, meaning that nouns tend to be polysemous (have more than one meaning), but with a low number of senses.

The relationship between the degree of polysemy and the occurrences of nouns in the OANC corpus is further illustrated in Figure 3.3b, which shows box plots for the total number of WordNet senses on the x-axis and the occurrences in OANC on the y-axis. Nouns with a high degree of polysemy shows higher median occurrences and higher third quantiles when compared to those in the inter quantile range in Figure 3.3a. In my opinion, the reason could be that more *senses* are indicative of more possible contexts in which a noun can occur. Similar to Figure 3.3b, Figure 3.3c and Figure 3.3d shows how polysemy is distributed across the OANC corpus, but this time they differentiate singular from plural nouns. By comparing both, it can be noticed that nouns in the singular occur more often than in plurals. A simple explanation could be that mass and count nouns can occur both in singular, but mass nouns cannot occur in plural.

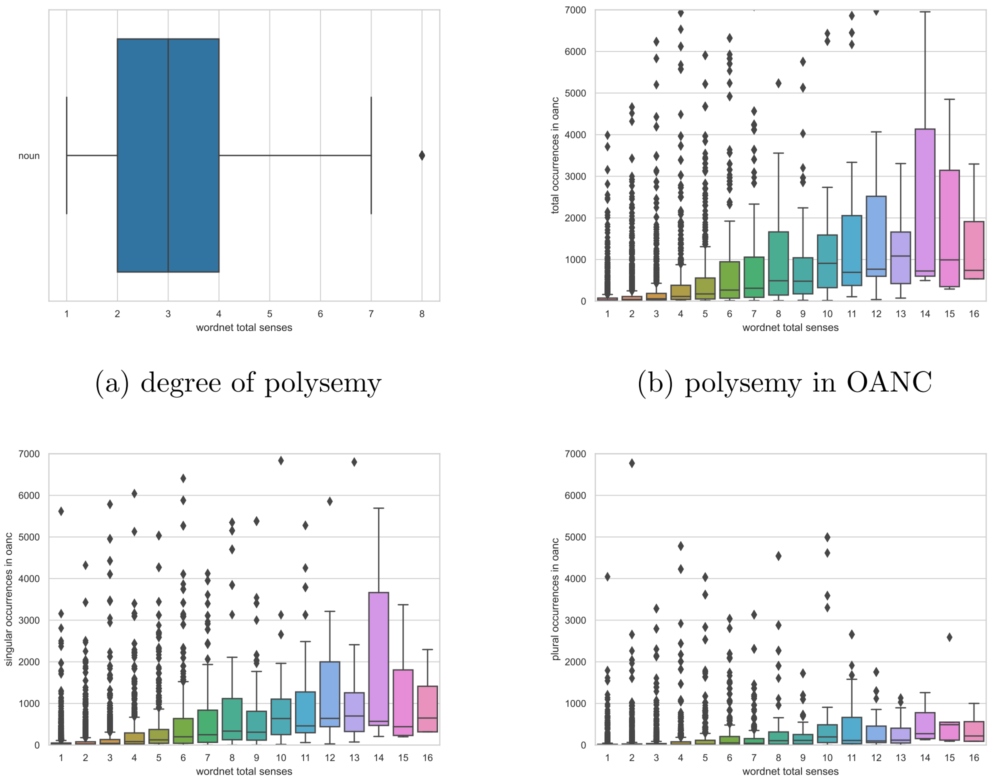
Example:

i. John purchases merchandise$_{\texttt{+sg +mass}}$ from Star Wars.

ii. !John purchases merchandise<u>s</u>$_{!\texttt{+pl +mass}}$ from Star Wars.

iii. John purchases a sweatshirt$_{\texttt{+sg +count}}$ from Star Wars.

iv. John purchases some sweatshirts$_{\texttt{+pl +count}}$ from Star Wars.



(a) degree of polysemy

(b) polysemy in OANC

(c) polysemy in OANC (singular)

(d) polysemy in OANC (plural)

Figure 3.3: Distribution of polysemy in BECL 2.1.

3.1.2 The Brysbaert Concreteness Ratings

The *"Concreteness ratings for 40 thousand generally known English word lemmas"* Brysbaert et al. (2013) provides the gold standard for the *degree of abstractness* of nouns. Words were graded on a concreteness scale from 1 to 5, with 1 being the minimum and 5 the maximum. The annotations provide a mean of the concreteness ratings for each word. This thesis refers to the mean value of the ratings as *"degree"*. Because the focus is on nouns, the dataset can be down scaled by filtering out all the rest. The *Institute for Natural Language Processing* (IMS) of the University of

Stuttgart provides a subset of the norm with Part-Of-Speech tags that facilitates this task (Tater et al., 2022).

Because in this prefers using the term *abstractness* over *concreteness*, there is the need to map *concreteness degrees* to *abstractness degrees* for coherency. To archive that goal, a function f takes a *concreteness degree* x as an argument and returns an *abstractness degree* y. The function f first subtracts the *concreteness degree* x from the lower *concreteness degree* possible `MIN_DEG` (here 1) and later adds the higher *concreteness degree* possible `MAX_DEG` (here 5).

Function: `concreteness` $\mapsto$ `abstractness`

$$f(x) = (\texttt{MIN\_DEG} - x) + \texttt{MAX\_DEG} = y$$

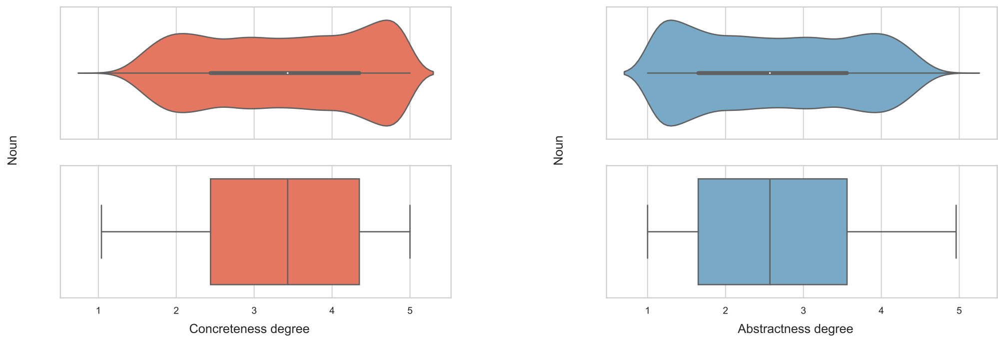
$$x : \textit{concreteness degree}$$

$$y : \textit{abstractness degree}$$

Word	...	Conc.M	Conc.SD	...	ENCOW-POS	ENCOW-FREQ
I	...	3.93	1.44	...	PP	78604541
a	...	1.46	1.14	...	DT	188935216
aardvark	...	4.68	0.86	...	NN	1317
aback	...	1.65	1.07	...	ADV	11577
abacus	...	4.52	1.12	...	NN	2659
...	...	...	...	...	...	...
zoophobia	...	2.04	1.02	...	NN	9
zucchini	...	4.87	0.57	...	NN	8523

Table 3.3: Snippet of Brysbaert Concreteness Ratings (Filtered Part-Of-Speech)

The Brysbaert norm contains 17115 nouns. Figure 3.4a and Figure 3.4b shows (a) the degree of concreteness and (b) the degree of abstractness of *rigid* nouns. Figure 3.4b presents visually how the function that maps *concreteness ratings* into *abstractness ratings* reversed the *concreteness* values to fit the definition of *abstractness*. Figure 3.4a and Figure 3.4b illustrates a violin and a box plot. On the axis are placed the *degree* (*concreteness* or *abstractness*) and on the y-axis the nouns. Because the two graphs describe the same phenomenon, from now on, the thesis will refer only to the degree of abstractness (Figure 3.4b). Figure 3.4b shows that the median degree of abstractness is approximately 2.5, with an interquartile range between 1.5 and 3.5. Furthermore, Figure 3.4b shows that a large portion of nouns have a low degree of abstractness, with only a few having a degree higher than 4.



(a) Degree of Concreteness

(b) Degree of Abstractness

Figure 3.4: Distribution of Degrees in The Brysbaert's Norm

3.1.3 The Abstractness Degrees of Rigid Nouns

The concreteness ratings provided in Brysbaert et al. (2013) are not available for *word-senses*. For this matter, it is not possible to test on *elastic* nouns if *mass senses* tend to be more abstract than *mass senses* (H1). For *rigid* nouns, since they are *mass* or *count* only, the granularity of the data of the BECL annotation can be reduced from *noun-senses* to nouns, without losing information on countability status. Figure 3.5a shows the occurrences of *countability classes* in BECL 2.1 on the x-axes and the *countability classes* (*mass* and *count*) for rigid nouns on the y-axis. Comparing it to Figure 3.2 it can be noticed that the distribution of countability in *rigid* nouns is almost identical to the distribution in the whole BECL 2.1 corpus.

The box plot and the violin plot in Figure 3.5c and Figure 3.5e present the *countability classes* on their x-axis and the degree of abstractness of the *rigid mass* and *rigid count* nouns on the y-axis. These graphs show the tendency of *rigid* mass nouns being more abstract than *rigid* count nouns and 'vice versa'. This confirms (H1), mass nouns, as they typically denote substance (section 2.4), tend to be more abstract than count nouns.

Before proceeding with the extraction of features from a corpus to make nouns' representations, the baseline of the gold standard needs to be balanced. Training a model with an unbalanced dataset has an impact on the accuracy score. For example, if 90% of nouns are *count* and only 10% are *mass*, then a model could classify all nouns as being *count* and still achieve a score of .90. To balance the datasets, the number of count nouns was reduced by deleting them randomly. After balancing the baseline, the distribution of the abstractness degrees in *rigid* mass and count nouns retained their initial proportions. The Figure 3.5d and Figure 3.5f are almost identical with Figure 3.5c and Figure 3.5e. This is important, then mass nouns should have the tendency to be more abstract than count nouns (H1) in the balanced standard as well.

The Standard. The dataset provides for every *rigid* nouns, the token (the noun itself), the *countability class* ('mass' or 'count'), the *abstractness degree* (scalar from 1 to 5), the *abstractness rate* (a whole number from 1 to 5) and the *abstractness class* ('abstract', 'concrete' or '-'). The *abstractness rates* were obtained by rounding the *abstractness degrees* and are used to train and evaluate a model for multi-class classification task to try automatic rate the abstractness of nouns. The *abstractness classes* were obtained by mapping nouns rated with an *abstractness* of '1' or '2' as 'concrete' and those rated with '4' or '5' as 'abstract'. Nouns with an abstractness rate of 3 were not assigned to a class ('-'). The *abstractness classes* are utilized to train and evaluate a binary model that classifies *rigid* nouns as being either abstract or concrete.

nn	count.cls	abst.deg	abst.rate	abst.cls
abscess	count	1.52	2	concrete
abyss	count	2.93	3	-
acceptability	mass	4.26	4	abstract
...	...	...	...	...
yoga	mass	1.47	1	concrete
zoning	mass	3.55	4	abstract

Table 3.4: Snippet of The Gold Standard

Gold-Standard: Abstractness Degree of Rigid Nouns

#1 `Noun` (token)

#2 `Countability Class`

#3 `Abstractness Degree`

#4 `Abstractness Rate` (rounded degree)

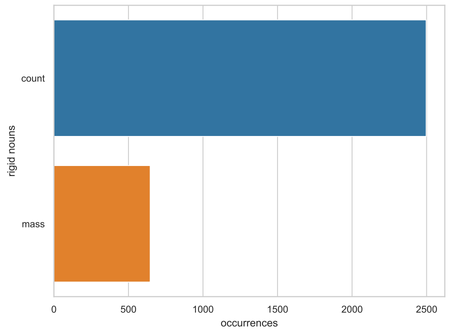
#5 `Abstractness Class`

Function: abstractness `rate` ↦ `class`

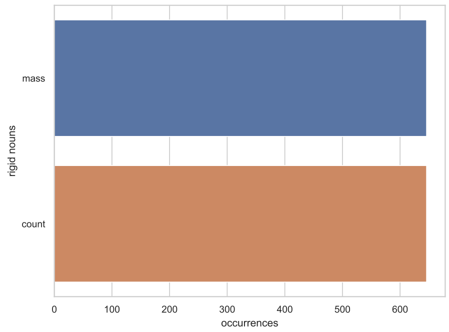
$\boxed{1}$ ***or*** $\boxed{2}$ ↦ $\boxed{\texttt{concrete}}$

$\boxed{4}$ ***or*** $\boxed{5}$ ↦ $\boxed{\texttt{abstract}}$

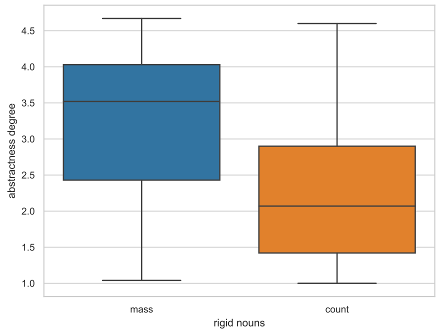
$\boxed{3}$ ↦ $\boxed{\texttt{-}}$



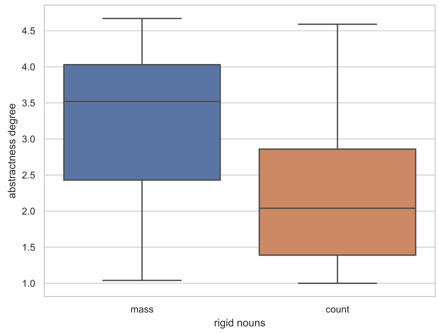
(a) Nouns' Distribution (unbalanced)



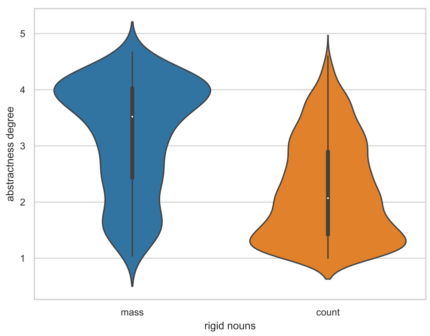
(b) Nouns' Distribution (balanced)



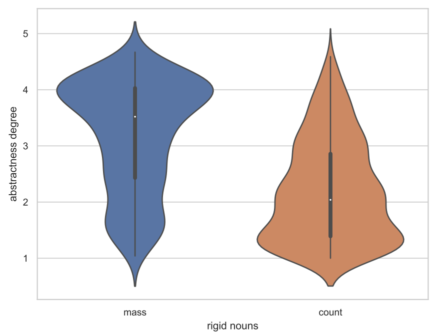
(c) Abstractness' Distribution (unbalanced)



(d) Abstractness' Distribution (balanced)



(e) Abstractness' Distribution (unbalanced)



(f) Abstractness' Distribution (balanced)

Figure 3.5: Abstractness Degrees of Rigid Nouns

3.1.4 Corpora from the Web (COW)

To provide nouns' representation, this thesis utilizes the ENCOW (Schaefer, 2015) corpus to extract features that describe the mass-count distinction. This section provides the list of information that can be extracted from the corpus, along with some examples on how the information could be potentially utilized.

The ENCOW Corpus provides following informations about the tokens:

(1) `token`: literal token

> **Example:** counting occurrences of the target noun with literal tokens.
> i. Laura purchased a *sweatshirt*$_{\texttt{target}}$.
>
> ii. Laura purchased some *merchandise*$_{\texttt{target}}$.

(2) `tag`: Part-Of-Speech tag for this token

> **Example:** identify determiners-nouns compounds.
> i. Laura purchased a(`DT`) *sweatshirt*$_{\texttt{target}}$(`NN`).
>
> ii. Laura purchased some(`DT`) *merchandise*$_{\texttt{target}}$(`NN`).

(3) `lemma`: lemma for this token

> **Example:** identify the lemma of the target noun.
> i. Laura purchased a *sweatshirt*$_{\texttt{target}}$. (`lemma=sweatshirt`)
>
> ii. Laura purchased two *sweatshirts*$_{\texttt{target}}$. (`lemma=sweatshirt`)

(4) `named-entity`: named entity label for this token

> **Example:** identify an *entity*.
> i. *Laura*$_{\texttt{PERSON}}$ purchased a sweatshirt.
>
> ii. Laura lives in *London*$_{\texttt{LOCATION}}$.

(5) `tag-simple`: simplified tag

> **Example:** identify determiners-nouns compounds (with a simplified version of the tag).
> i. Laura purchased a(`D`) *sweatshirt*$_{\texttt{target}}$(`N`).
>
> ii. Laura purchased some(`D`) *merchandise*$_{\texttt{target}}$(`N`).

(6) `morphology`: morphological attributes of this token

> **Example:** retrieve information about the *numerous* of the target noun.
> i. Laura's *girlfriend*$_{\texttt{target}}$(`+singular`) purchased a sweatshirt.
>
> ii. Laura purchased two *sweatshirts*$_{\texttt{target}}$(`+plural`).

(7) `index`: this token's running index in this sentence

> **Example:** identify the position of the target noun in the sentence.
> i. Laura's *girlfriend*$_{\texttt{target}}$(`index=2`) purchased a sweatshirt.
>
> ii. Laura purchased two *sweatshirts*$_{\texttt{target}}$(`index=4`).

(8) `head-index`: index of this token's dependency head

> **Example:** identify the position of the target noun's head in the sentence.
> i. Laura's sweatshirt is part of(`head`) a *collection*$_{\texttt{target}}$(`head-index=5`).
>
> ii. Laura's sweatshirt$_{\texttt{target}}$(`head-index=3`) is(`head`) red.

(9) `relation`: dependency relation between this token and its head

> **Example:** know which relation the target noun has with its head.
> i. Laura's sweatshirt is part of(`head`) a *collection*$_{\texttt{target}}$(`rel=probj`).
>
> ii. Laura's sweatshirt$_{\texttt{target}}$(`rel=subj`) is(`head`) red.

token	tag	lemma	named-entity	tag-simple	morphology	index	head-index	relation
...	...	...	...	...	...	...	...	...
Now	RB	now	O	C	—	1	5	advmod
,	,	,	O	c	—	2	5	punct
The	DT	the	O	D	—	3	4	dep
Putter-Awayer	NP	(unknown)	O	N	sg	4	5	nsubj
has	VBZ	have	O	V	ind—pres—3—sg	5	0	null
the	DT	the	O	D	—	6	9	det
steepest	JJS	steep	O	A	sup	7	9	amod
learning	NN	learning	O	V	part—pres	8	9	nn
...	...	...	...	...	...	...	...	...

Table 3.5: Snippet of ENCOW

3.2 Features Extraction

Training a machine learning model requires a numerical representation of the words, in this case nouns. The word-vectors employed in this thesis contains a set of features that provide information about the countability status of the nouns. Thanks to the literature discussed in chapter 2, six sets of features were extracted from the nouns in the gold standard from the ENCOW corpus.

The vector $\vec{V}_{\text{Features}}$ refers to the *union* of all sets of features from $\vec{V}_1$ to $\vec{V}_2$. This set contains all the information extracted.

Features' vector

$$\vec{V}_{\text{Features}} = \bigcup_{i=1}^{6} \vec{V}_i = \vec{V}_1 \cup ... \cup \vec{V}_6$$

3.2.1 $\vec{V_1}$ – Pluralization

$$\vec{V_1} =< \#nn,\ \#sg,\ \#pl >$$

Assumption: Only count nouns can be pluralized (Pelletier, 2012).

Features: It keeps track of the frequency with which a noun is observed within the corpus ($\#nn$), whether it is in its singular form ($\#sg$) or in its plural form ($\#pl$).

Examples:

i.	Maria eats one *apple*.	The desk is considered *furniture*.
ii.	Maria eats two *apples*.	!Desks are considered *furnitures*

3.2.2 $\vec{V}_2$ – Dependency relation between a noun and its head

$$\vec{V}_2 =< \#\textit{rel:type}_1,\ ...,\ \#\textit{rel:type}_n >$$

Assumption: Certain types of dependency relations may be more prevalent with count nouns than with mass nouns, and vice versa (Ware, 1979).

Features: It keeps track of the frequency with which several types of dependency relations ($\#\textit{rel:type}_i$) between a noun and its head are observed within the corpus.

Examples:

i. ... was part of a major *plan* ... <*head:of*, *rel:pobj*>

ii. ... validating the *accuracy* of ... <*head:validating*, *rel:dobj*>

3.2.3 $\vec{V}_3$ – Noun's head

$$\vec{V}_3 = < \#head{:}word_1,\ ...,\ \#head{:}word_n >$$

Assumption: Information about countability can be provided by the head of a noun.

Features: It keeps track of the frequency with which a word is observed as the head of a noun ($\#head{:}word_i$).

Examples:

i. ... was part of a major *plan* ... <*head:of*>

ii. ... validating the *accuracy* of ... <*head:validating*>

3.2.4 $\vec{V}_4$ – Part-of-Speech tag on the noun's head

$$\vec{V}_4 =< \#\textit{head-tag:tag}_1,\ ...,\ \#\textit{head-tag:tag}_n >$$

Assumption: Information about countability can be provided by the head's Part-of-Speech Tag of a noun (Pelletier, 2012).

Features: It keeps track of the frequency with which a Part-of-Speech Tags is observed as the tag for the head of a noun ($\#\textit{head-tag:tag}_i$).

Examples:

i. ... was part of a major *plan* ... <*head:of*, *head-tag:IN*>

ii. ... validating the *accuracy* of ... <*head:validating*, *head-tag:VBG*>

3.2.5 $\vec{V}_5$ – Preposition "of" as noun's head

$$\vec{V}_5 =< \#head{:}of >$$

Assumptions: The grammatical distinction between mass and count reflects the ontological distinction between individuals and substance. If *"of"* is the noun's head, this noun should denote substance and be a mass noun (Pelletier, 2012).

Features: It keeps track of the frequency with which the preposition "*of*" is observed as the head of a noun (*#head:of*).

Examples:

i. *bottle* ***of*** *water* <*individual/count*> ***of*** <*substance/mass*>

bottle$<$*head:of* $= 0>$ *water*$<$*head:of* $= 1>$

ii. *liters* ***of*** *water* <*measure/count*> ***of*** <*substance/mass*>

liters$<$*head:of* $= 0>$ *water*$<$*head:of* $= 1>$

3.2.6 $\vec{V}_6$ – Appropriateness of noun's determiners

$$\vec{V}_6 :=< \#det{:}x_1,\ \ldots,\ \#det{:}x_n >$$

$x_i \in$ *articles* := {`the, a, an`} ∪
demonstratives := {`this, that, these, those, which`} ∪
possessive pronouns := {`my, your, our, their, his, hers, whose, its`} ∪
distributive words := {`all, both, half, either, neither, each, every`} ∪
quantifiers := {`much, little, some, most, more, few, several, certain, many, any, enough, no, none`} ∪
pre-determiners := {`such, what, rather, quite`} ∪
ordinals := {`first, second, third, next, last`}

Assumption: The appropriateness of a noun with a certain determiner is relevant for the distinction between count and mass. Count nouns are appropriate for *enumeratives*, while mass nouns are appropriate for *ammassives* (Ware, 1979).

Features: It keeps tracks of the frequency with which a determiner is observed preceding a noun ($\#det{:}x_i$).

Examples:

i.	There is so <u>much</u> *water*.	!There is so <u>much</u> *apple*.
ii.	!I drank my <u>first</u> *water*.	I ate my <u>first</u> *apple*.

3.3 Models

This thesis implements three classifiers based on a random forest of decision trees, implementing them with Scikit-learn (Pedregosa et al., 2011). The models (M_1, M_2 and M_3) were trained using 63 different combinations of features-vectors, resulting in a total of 189 trained models. The first model (M_1) predicts the *countability class* (*mass/count*) of the *rigid* nouns. The model M_2 predicts the *abstractness rate* (whole number from *1* to *5*) of a *rigid* noun. The third model (M_3) predicts the *abstractness class* (*abtract/concrete*) for *rigid* nouns that are considered to being either *abstract* or *concrete* (exluding nouns with an *abstractness rate* of *3*).

Models

M_1 – **Mass-Count Classifier**

Predicts the countability status of a noun.
Generalization on the degree of abstractness (scalar from 1 to 5).
Labels:={`mass`, `count`}

M_2 – **Abstractness Rate Predictor**

Predicts the abstractness rate of a noun.
Generalization on countability status (binary with mass or count).
Labels:={`1`, `2`, `3`, `4`, `5`}

M_3 – **Abstract-Concrete Classifier**

Predicts the abstractness class of a noun.
Generalization on countability status (binary with mass or count).
Labels:={`abstract`, `concrete`}

Random Forest Classifier and k-Folds Cross-Validation

The model implemented for the classification tasks is a *Random Forest Classifier* (RFC) from Scikit-learn (Pedregosa et al., 2011). The model fits a number (default=100) of *Decision Tree Classifiers* on various subsamples of the datasets and uses averaging to improve the predictive accuracy and control over-fitting.

Decision trees. A *Decision Tree Classifier* splits the predictor space of the target variable into more homogeneous sub-spaces. For example, by predicting the mass and count of rigid nouns, the tree splits its nodes into more homogeneous groups in therms of mass nouns or count nouns.

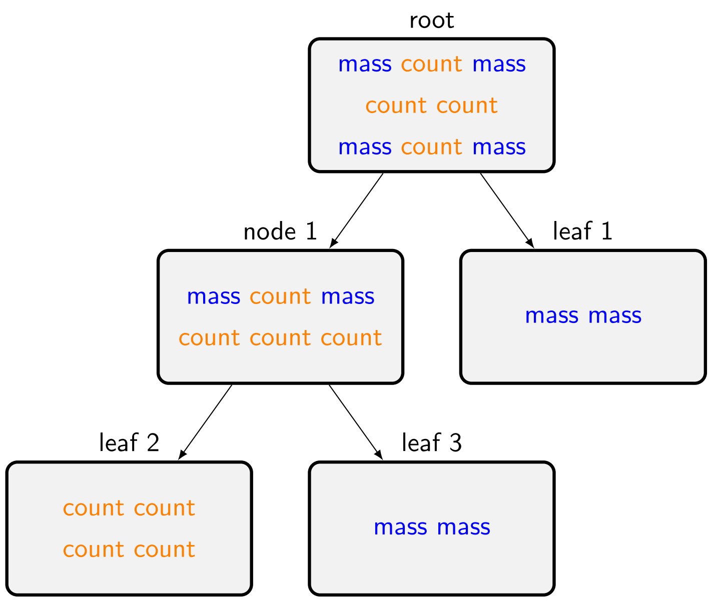


Figure 3.6: Decision Tree

Slitting Criterion. There are various ways to split a node, and they are divided into two categories based on the type of the target variable. For (1) *continuous target variables* the nodes are splitted by reducing variance, for (2) *categorical target variables* the nodes can be splitted by calculating the *Information Gain*, *Gini Impurity* or *Chi-Square*. The splitting criterion used by the models is the *Gini Impurity* and indicates the *impurity* of a node. Then, the lower the *Gini Impurity*, the lower the likelihood of misclassification (a pure node has an impurity of zero).

$$\textit{Gini Impurity} = 1 - \sum_{i=1}^{n} P(i)^2$$

$P(i)$: probability of seeing a class

n : number of classes

Example:

The *Gini Impurity* can be calculated for all nodes in Figure 3.6.

$$\textit{Gini Impurity} = 1 - (P(\texttt{+mass})^2 - P(\texttt{+count})^2)$$

$\textit{Gini Impurity}_{\texttt{root}} = 1 - ((\frac{1}{2})^2 + (\frac{1}{2})^2) = 0.5$

$\textit{Gini Impurity}_{\texttt{node 1}} = 1 - ((\frac{1}{3})^2 + (\frac{2}{3})^2) = 0.\overline{4}$

$\textit{Gini Impurity}_{\texttt{leaf 1}} = 1 - ((1)^2 + (0)^2) = 0$ (**pure node**)

$\textit{Gini Impurity}_{\texttt{leaf 2}} = 1 - ((0)^2 + (1)^2) = 0$ (**pure node**)

$\textit{Gini Impurity}_{\texttt{leaf 3}} = 1 - ((1)^2 + (0)^2) = 0$ (**pure node**)

K-Folds Cross-validation. The models were validated with the *k-folds cross-validation* technique. The dataset is being splitted into k folds, and each fold is used once to validate and $k-1$ times to train the model. This thesis validates its models with a 10-folds.

Iteration 1	**Test**	Train	Train	Train
Iteration 2	Train	**Test**	Train	Train
...	...			
Iteration k	Train	Train	Train	**Test**

Figure 3.7: k-folds cross-validation

4 Results and Discussion

In this section, the results obtained by training and evaluating the models (M_1, M_2 and M_3) are being presented and discussed. To test if a model M, trained to classify mass and count nouns, can generalize on the degree of abstractness of those nouns (H2), a mass-count classifier M_1 with different sets of features. To test if the features used to train M_1 can be used to make predictions on *abstractness* (H3) and then generalize on the *countability status* of the target nouns (H4), the thesis trains the models M_2 to try to rate the *abstractness* of *rigid* nouns (from *1* to *5*) and the model M_3 to try making a binary classification of *rigid abstract* and *concrete* nouns.

4.1 Results

All models are evaluated by calculating the mean accuracy of the k-folds iterations. To determine if the models generalize on the *abstractness* (Mass-Count Classifier M_1) or on the *countability* (Abstractness Rate Predictor M_2 and Abstract-Concrete Classifier M_3) of the predicted nouns, the gold standard is compared with the predicted labels aligned to the expected *abstractness* (Mass-Count Classifier M_1) or *countability* values (Abstractness Rate Predictor M_2 and Abstract-Concrete Classifier M_3).

4.1.1 Mass-Count Classifier (M_1)

The Mass-Count Classifier M_1 was trained in 63 different configurations, and all results can be found in the appendix A.1. In this section, I discuss only a few of those models, which results are also shown in Table 4.1.

rank	$\vec{V}_1$	$\vec{V}_2$	$\vec{V}_3$	$\vec{V}_4$	$\vec{V}_5$	$\vec{V}_6$	**acc**	**std**
34	X	X	X	X	X	X	0.8854	0.0223
2	X	-	-	-	-	X	0.9621	0.0149
16	-	-	-	-	-	X	0.9388	0.0198
24	X	-	-	-	-	-	0.9063	0.0241
55	-	X	-	-	-	-	0.8699	0.0424
62	-	-	-	X	-	-	0.8428	0.0263
63	-	-	-	-	X	-	0.6494	0.0337

Table 4.1: Slice of M_1 Evaluation Table (Appendix A.1)

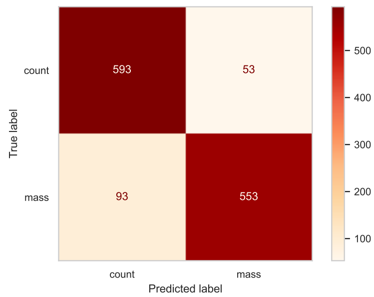
Training with all features

The Mass-Count Classifier $M_1(\vec{V}_{\text{Features}})$ achieved an accuracy score of 0.89, meaning that the classification can be considered as being reliable (Table 4.1). Figure 4.1c shows the distribution of abstractness in the gold standard. On the x-axis presets the *countability classes* and on the y-axis the *abstractness degrees* of the nouns. Figure 4.1d shows the degree of abstractness of the predicted targets. Comparing Figure 4.1d with the gold standard (Figure 4.1c) it is possible to state that the Mass-Count classifier M_1 trained with all features ($\vec{V}_{\text{Features}}$) is capable of generalizing on the degree of abstractness of the predicted nouns.

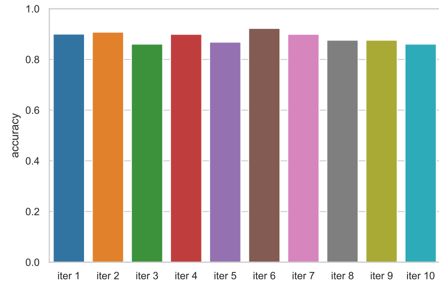
Higher scoring models

Several instances of the Mass-Count Classifier M_1 scores around 0.96, with only a marginal difference in accuracy to each other (Appendix A.1). Furthermore, it should be considered the fact, that the ranking of those instances could slightly change with a new training cycle, making it difficult to say which model is better than the other. Another interesting observation is that V_1 and V_6 are a component in the features' vectors with which all those models were trained.

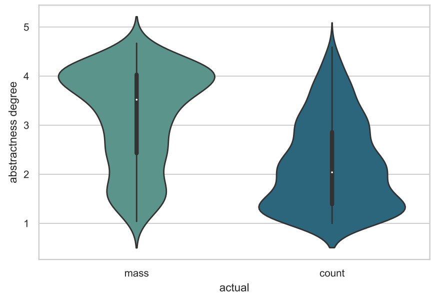
The Mass-Count Classifier $M_1(\vec{V}_1)$ and $M_1(\vec{V}6)$ performed good as well; the first achieved a score of 0.92 and the second a score of 0.94 accuracy. The model $M_1(\vec{V}_1 \cap$



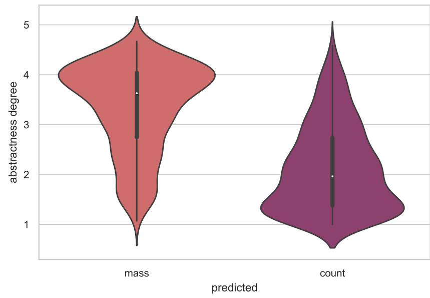
(a) confusion matrix



(b) scores



(c) distribution of abstractness (actual)

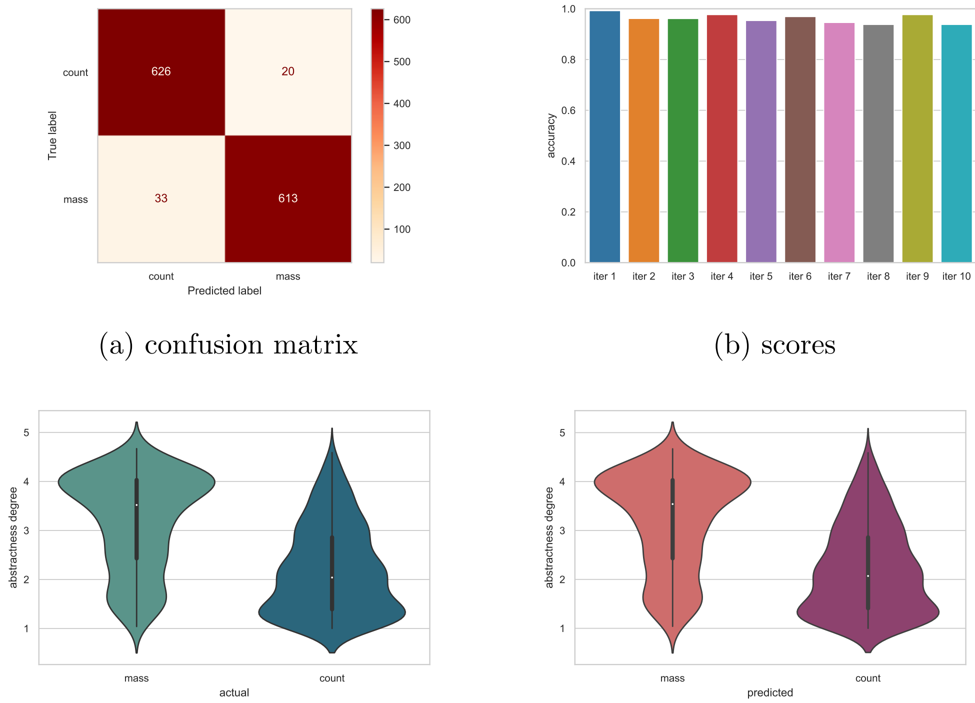


(d) distribution of abstractness (predicted)

Figure 4.1: Evaluation of Mass-Count Classifier $M_1(\vec{V}_{\text{Features}})$

$\vec{V}_6$) ranks at position 2 with an accuracy of 0.96 (Table 4.1). These results draw the conclusion that the features collected in vectors $\vec{V}_1$ and $\vec{V}_6$ could be the mainly responsible for the mass count distinction in *rigid* nouns. I think, however, that this could also be true for the *elastic* nouns. Then, (1) pluralization triggers countability shift in *elastic* nouns (Zamparelli, 2020) and (2) the appropriateness of a determiner with a mass or a count noun has more to do with its countability status than with the noun itself. However, this hypothesis needs to be investigated on the elastic nouns themselves.

Regarding how those high scoring models generalize on the degree of abstractness of the predicted nouns, they seem to generalize even better than $M_1(\vec{V}_{\text{Features}})$. Then, by looking at the distribution of abstractness in Figure 4.2d the *'belly'* around 1.5 *abstractnesses degree* (y-axis) by the predicted mass nouns, indicates that Mass-Count Classifier $M_1(\vec{V}_1 \cup \vec{V}_6)$ is doing a better job in classifying *concrete* mass, if compared to the $M_1(\vec{V}_{\text{Features}})$ (Figure 4.1d).



(a) confusion matrix

(b) scores

(c) distribution of abstractness (actual)

(d) distribution of abstractness (predicted)

Figure 4.2: Evaluation of Mass-Count Classifier $M_1(\vec{V}_1 \cup \vec{V}_2)$

Lower scoring models

The lower scoring Mass-Count Classifier is $M_1(\vec{V_5})$, presenting only an accuracy of 0.65 (Table 4.1). This result is to be expected, then, the vector $\vec{V_5}$ contains only information about nouns to be headed by the preposition "*of*" (Section 3.2.5). The model does not generalize on the abstractness degrees of the predicted nouns as well as the former ones. Then, it is classifying too many *concrete* nouns as being *mass* and too many *abstract* nouns as being *count*. This issue is made visible by the silhouette of the violin plot in Figure 4.3d.

False values in $M_1(\vec{V_1} \cup \vec{V_6})$

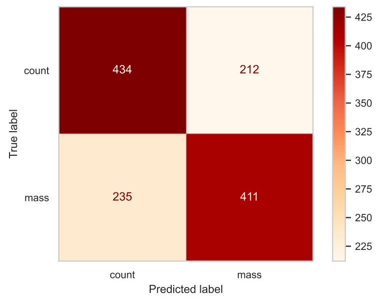
Figure 4.2a shows a confusion matrix for the evaluation of the Mass-Count Classifier $M_1(\vec{V_1} \cup \vec{V_6})$. The x-axis presents the *predicted* labels and the y-axis the *actual* labels. By analyzing the misclassified nouns, it can be observed that in the list of *false mass* many of the nouns refer to *general* but kind of concrete concepts (e.g. counterculture, statement, explosive) and many nouns found in *false count* refer to *specific* but kind of abstract ones (e.g. dementia, methodology, measurement). However, it can be challenging to discern between abstract/general and specific/concrete concepts.

False Mass

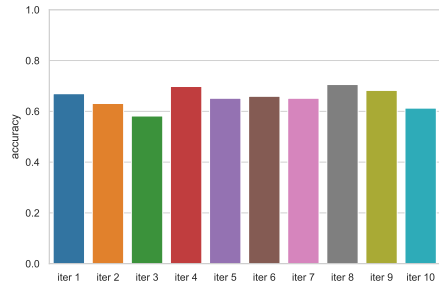
wage, cervix, confluence, docudrama, playbook, counterculture, handover, inaugural, resale, gallbladder, wacko, artifice, forceps, hinge, vantage, ante, understatement, cleansing, downtown, explosive

False Count

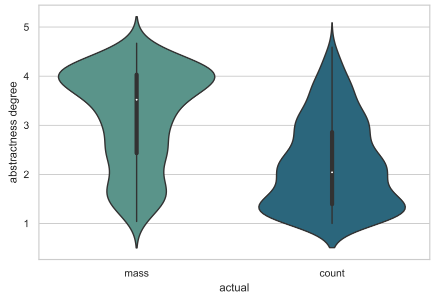
clientele, peroxide, retail, dementia, southward, woe, kinetics, methodology, render, uptown, procurement, therapeutics, artwork, polymorphism, analgesic, cum, evil, inflow, velvet, anesthetic, drool, health, litmus, parkland, spelling, spillover, beating, finance, help, legality, measurement, scat, siding



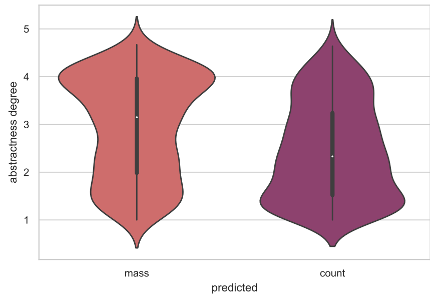
(a) confusion matrix



(b) scores



(c) distribution of abstractness (actual)



(d) distribution of abstractness (predicted)

Figure 4.3: Evaluation of Mass-Count Classifier $M_1(\vec{V}_5)$

Findings

After training and testing of several instances of the Mass-Count Classifier (M_1), the results indicate that the hypothesis (H2) can be accepted. Then, a model $M := M_1$ was capable of making reliable prediction on the countability status of *rigid* nouns and to generalize on the abstractness degrees of the predicted nouns.

4.1.2 Abstractness Rate Predictor (M_2)

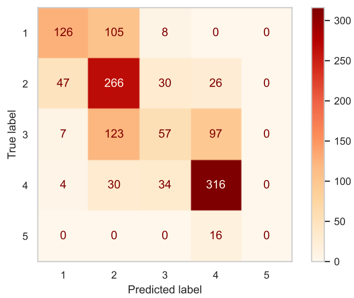
The Abstractness Rate Predictor M_2 was trained in 63 different configurations, and all results can be found in the appendix A.2. In this section, I discuss only a few of those models, which results are also shown in Table 4.2.

rank	$\vec{V}_1$	$\vec{V}_2$	$\vec{V}_3$	$\vec{V}_4$	$\vec{V}_5$	$\vec{V}_6$	acc	std
21	X	X	X	X	X	X	0.5914	0.0336
1	-	X	X	X	X	X	0.6107	0.0518
62	X	-	-	-	-	-	0.3584	0.0366
63	-	-	-	-	X	-	0.3213	0.0361

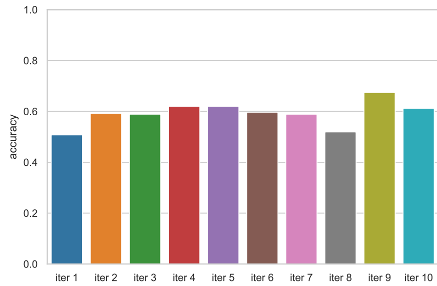
Table 4.2: Slice of M_2 Evaluation Table (Appendix A.2)

Models performance

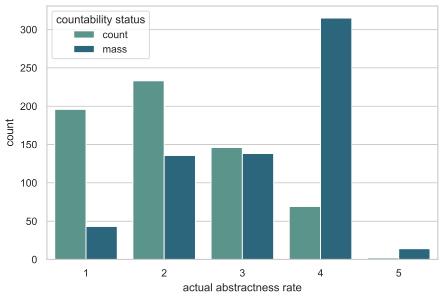
The higher scoring instance of the Abstractness Rate Predictor M_2 scores only with an accuracy of 0.61 (Table 4.2). The confusion matrix for $M_2(V_{\text{Features}})$ in Figure 4.4a shows the predicted labels on the x-axis and the actual labels on the y-axis. The two *'red spots'* on the diagonal of the matrix suggests that $M_2(V_{\text{Features}})$ is doing a better job at classifying rates *2* and *4*. Figure 4.4c shows the gold standard for the distribution of countability through the *abstractness rate*. The *abstractness rates* are displayed on the x-axis and the occurrences of the rating on the y-axis. For every rate, the plot shows two bars, one for each *countability class*. Looking at how the model generalizes to the countability status of the rigid nouns (Figure 4.4d) we notice that the count plot is similar to the gold standard (Figure 4.4c), The main difference can be noticed at the rate of *3*. Even if the proportions between mass and count resembles as the gold standard, more than the half of the nouns are missing (Figure 4.4d). At the same time, predicted count nouns with an abstractness rate of *2* and predicted mass nouns with an abstractness rate of *4* increased in comparison with the gold standard. This result cannot confirm the hypothesis (H5), but does not completely falsify it either.



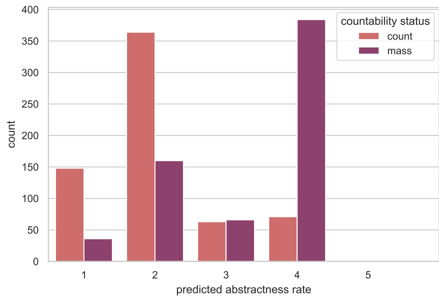
(a) confusion matrix



(b) scores



(c) distribution of countability (actual)



(d) distribution of countability (predicted)

Figure 4.4: Evaluation of Abstractness Rate Predictor $M_2(\vec{V}_{\text{Features}})$

Findings

The results achieved by all the instances of the Abstractness Rate Predictor (M_2) show that the model $M' :=$ Abstractness Rate Predictor M_1 is incapable of providing reliable ratings for the *rigid* nouns (H1) and is also incapable to generalize on the countability status of those (H2).

Because the models seem to perform better classifying *abstract* and *concrete* nouns, hypotheses (H3) and (H4) should be tested on a third model M'' to see if a (binary) Abstract-Concrete Classifier M_3 can reliably predict the *abstractness class* of *rigid* nouns and generalize to their *countability status*.

4.1.3 Abstract-Concrete Classifier (M_3)

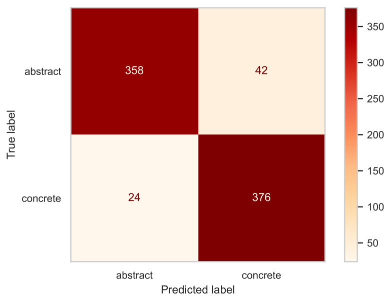
The Abstract-Concrete Classifier M_3 was trained in 63 different configurations, and all results can be found in the appendix A.3. In this section, I discuss only a few of those models, which results are also shown in Table 4.3.

rank	$\vec{V}_1$	$\vec{V}_2$	$\vec{V}_3$	$\vec{V}_4$	$\vec{V}_5$	$\vec{V}_6$	**acc**	**std**
12	X	X	X	X	X	X	0.9238	0.0253
1	-	X	X	X	X	X	0.9325	0.0211
36	-	-	-	-	-	X	0.8725	0.0215
40	X	-	-	-	-	X	0.8662	0.025
62	X	-	-	-	-	-	0.7025	0.0515
63	-	-	-	-	X	-	0.6225	0.0236

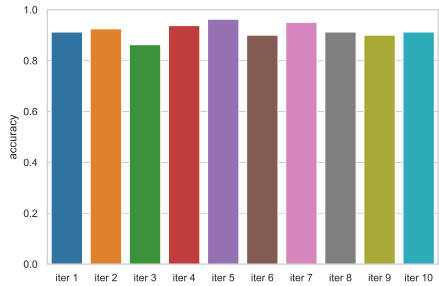
Table 4.3: Slice of M_3 Evaluation Table (Appendix A.3)

Training with all features

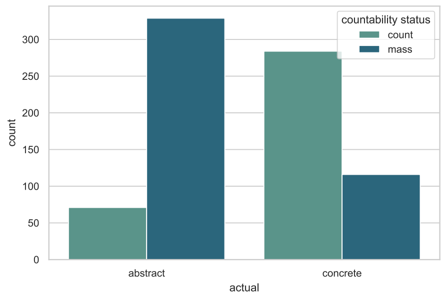
The Abstract-Concrete Classifier $M_3(\vec{V}_{Features})$ achieves a mean accuracy score of 0.92. Even if this model ranks 12th, the higher scoring model $M_3(\vec{V}_{Features} \setminus \vec{V}_1)$ scores less than 1% better with an accuracy of 0.93 (Table 4.3). Figure 4.5c shows the gold standard for the distribution of countability through the *abstractness classes*. The *abstractness classes* are displayed on the x-axis and the occurrences of the rating on the y-axis. For every *abstractness class*, the plot shows two bars, one for each *countability class*. Figure 4.5d shows the distribution of countability through the *abstractness classes* for the predicted nouns of the Abstract-Concrete Classifier $M_3(\vec{V}_{Features})$. The model was capable to generalize on the *countability status* of the predicted nouns. Comparing it to the gold standard in Figure 4.5c the bar plot in Figure 4.5d, it seems that $M_3(\vec{V}_{Features})$ classify some of the actual *abstract* count nouns as being *concrete*.



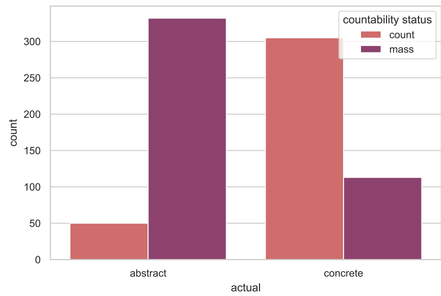
(a) confusion matrix



(b) scores



(c) distribution of abstractness (actual)

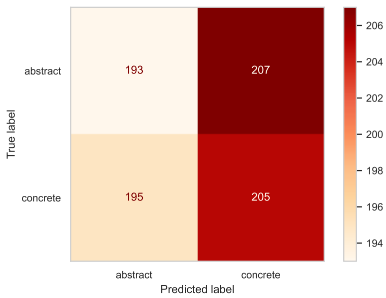


(d) distribution of abstractness (predicted)

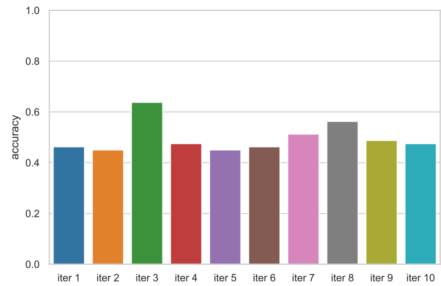
Figure 4.5: Evaluation of Abstract-Concrete Classifier $M_3(\vec{V}_{\text{Features}})$

Lower scoring models

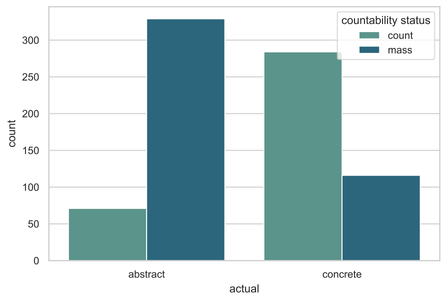
The lower-scoring Abstract-Concrete Classifier is $M_3(\vec{V}_5)$ which is consistent with the result obtained by the Mass-Count Classifier M_1 (subsection 4.1.1). More interestingly, the second lower scoring model is $M_3(\vec{V}_1)$ with an accuracy of 0.70 (Table 4.3). This suggests that pluralization is not a reliable feature to predict the abstractness class of a *rigid* noun. In contrast, the model $M_3(\vec{V}_6)$, although is not one of the higher-scoring models, scores for an accuracy of 0.87, which is considered a good score. This suggests that the determiners, that occur with a noun, may contain some information regarding its abstraction.



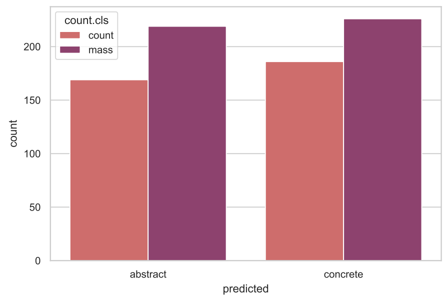
(a) confusion matrix



(b) scores



(c) distribution of countability (actual)



(d) distribution of countability (predicted)

Figure 4.6: Evaluation of Abstract-Concrete Classifier $M_3(\vec{V}_5)$

False values in $M_3(\vec{V}_{Features})$

By analyzing the misclassified nouns of the model $M_3(\vec{V}_{\text{Features}})$, it can be observed that 15 out of 24 *false abstract* predicted targets are mass nouns, and 30 out of 42 *false concrete* predicted targets are count nouns. Since the features were extracted to provide information about the mass-count distinctions of the targets, it is reasonable that the model tends to classify mass nouns as being *abstract* and count nouns as being *concrete*.

False Abstract

album, drool, fighting, jujitsu, riches, caricature, polygraph, sedimentation, waiver, hypertext, baldness, masturbation, eyesight, intercourse, laughter, cinematography, environmentalist, advertising, bedrock, thesis, czar, dunk, newsman, paralysis

False Concrete

reconnaissance, vantage, exploit, startup, conditioning, conformist, enhancer, fun, megalomaniac, minimum, miracle, prep, primary, throwback, essential, lowbrow, pun, dissident, goody, baffle, contingent, entire, oxygenation, refining, resale, accolade, advisory, rogue, spoof, microscopy, partisan, precursor, render, trusty, assistance, dementia, listening, motif, spiritual, brief, inaugural, processing

Findings

The results obtained by several of the instances of the Abstract-Concrete Classifier (M_3) show that even if a model M' cannot reliably predict the abstractness rate of a *rigid* noun, a binary classifier $M'' :=$ Abstract-Concrete Classifier M_3 can reliably classify a *rigid* noun being *abstract* or *concrete* (H3). Furthermore, M'' can generalize on the countability status of the predicted nouns (H4).

4.2 Discussion

The thesis aims to answer two questions: (Q1) what features should be extracted from a corpus to better describe the distinction between mass and count in English nouns? And (Q2) Can the same features extracted from a corpus to describe the mass-count distinction in English nouns be suitable to describe the degree of abstractness of those nouns? And with which resolution (binary or multi-class)? To answer those questions, this thesis tests several hypotheses thanks to three classification models based on the sci-kit learn implementation of the Random Forest Classifier.

First, in subsection 3.1.3 it was tested and confirmed that mass nouns, as they usually denote substance, tend to be more abstract than count nouns (H1). In section 3.2 several features sets were extracted from the ENCOW Corpus (Schaefer, 2015) to train the models.

The first model to be trained was the Mass-Count Classifier M_1 in order to test if a model M, trained to classify mass and count nouns, can generalize on the degree of abstractness of those nouns (H2). The results after training 63 instances of the models confirmed (H2) and show that several of them achieved high accuracy scores and are capable to generalize on the *abstractness* of the predicted nouns (subsection 4.1.1). The two features sets that this work identifies as being the most descriptive for the mass-count distinction are $\vec{V_1}$ – *pluralization*(subsection 3.2.1) and $\vec{V_6}$ – *appropriateness of determiners* (subsection 3.2.6). Then *pluralization* is a phenomenon that occurs with count nouns only, meaning that if in the corpus a noun is not being seen in plural form should be a *mass* nouns (Pelletier, 2012). Similar to the previous phenomenon, a *rigid* noun should only occur with those determiners that are *appropriate* for the corresponding *countability class* (Ware, 1979).

The second model to be trained was the Abstractness Rate Predictor M_2 to test if a model M', trained with the same features as M, is capable of predicting *abstractness ratings* (H3) and to generalize on the *countability status* of the target nouns. The results after training and evaluating 63 of the models did not fulfill the requirements to accept (H3) and (H4). The higher-scoring instance of the model scored only 0.61 (Table 4.2) and did not generalize well on the *countability class* of the targets (subsection 4.1.2). The reason why the models fail in making a multi-class prediction could be the fact that the features were collected with the binary distinction of *mass* and *counts* in mind. Furthermore, the higher-scoring M_2 model instance was doing a better job classifying *abstract* and *concrete* nouns and struggled more on those nouns with a rate of *3* (subsection 4.1.2). These results led to the decision to train a third, this time binary, model M_3, an Abstract-Concrete Classifier.

With the Abstract-Concrete Classifier M_3, this thesis tests if a model M'', trained with the same features as M and M', is capable of predicting *abstractness class* (H3) and to generalize on the *countability status* of the target nouns (H4). This time, the results show several models M_3 with a high accuracy score. Those models are capable to generalize on the *countability status* of the predicted nouns (subsection 4.1.3). With these results, the hypotheses (H3) and (H4) can be partially accepted. Taking a look a how the features sets performed, it is interesting to notice that $\vec{V}_1$ – *pluralization* (subsection 3.2.1) is the worse performing features set and lower the score of almost all instances of M_3 where this set is present as a part of the features (Appendix A.3), suggesting that the phenomenon of *pluralization* in nouns is related to the mass-count distinction only. Instead, the features set $\vec{V}_6$ – *appropriateness of determiners* performed quite well, with the Abstract-Count Classifier $M_3(\vec{V}_6)$ scoring at 0.87 (Table 4.3), suggesting that maybe some determiners are more appropriate with *abstract* and others with *concrete* nouns. The best scoring instance of the M_3 is $M_3(\vec{V}_{Features} \setminus \vec{V}_1)$ with a score of 0.93 (Table 4.3), suggesting that a model to *understand* the abstract-concrete distinction may need additional information about the nouns that for the mass-count distinction is not required.

After having tested all the hypotheses and having discussed the results, an answer to the research question can be provided. The first question that the thesis tries to answer is what features should be extracted from a corpus to better describe the distinction between mass and count (Q1). Based on efficiency and quality of result, the features of the vectors $\vec{V}_1$ – *pluralization* and $\vec{V}_6$ –*appropriateness of determiners* performed better than the other to describe the mass-count distinction (A1). The second question is more complex to be answered. First, it must be said that a multi-class classification in not possible utilizing the feature that this thesis extracted for the mass-count distinction. Then, the features extracted are capable only of making a binary classification of *abstract* and *concrete* nouns. The answer to which are the best performant features sets is not straightforward, and it depends on the use cases. To obtain the best scoring possible, the answer is to extract all features of section 3.2, but not $\vec{V}_1$ – *pluralization* (subsection 3.2.1). To be more efficient, $\vec{V}_6$ – *appropriateness of determiners* (subsection 3.2.6) only (A2). Having to deal with only the small set of features of the vector $\vec{V}_6$ as some advantages, it not only makes *features extractors* (scripts that extract features from the corpus) easier to implement, but it also uses less resources (storage and processing power).

(A1) The features to extract from a corpus that better describes the mass-count distinction in English rigid nouns are those concerning *pluralization* of nouns and the *appropriateness of determiners* with nouns.

(A2) The features that describe the English mass-count distinction in English rigid nouns and are suitable to describe the *abstractness* are those concerning the *dependency relation between the noun and its head*, the *token* and *Part-Of-Speech tag of the noun's head*, and the *appropriateness of determiners* with the nouns. The most efficient features to extract from a corpus, is the *appropriateness of determiners* with the nouns. The feature is only capable of describing *abstractness* as a binary distinction between *abstract* and *concrete* nouns.

Future Work

This thesis examines the abstractness of rigid nouns by evaluating three models (Mass-Count Classifier M_1, Abstractness Rate Predictor M_2 and Abstract-Concrete Classifier M_3) and making them generalize ether on the *abstractness* or on the *countability status* of the nouns on which it made a prediction. Future work should investigate further on the abstractness degrees of *elastic* nouns and their *senses* (polysemy). For that reason, there is the need for a new annotation to provide *abstractness-/concreteness ratings* for *word-senses* (in WordNet). An idea to collect these ratings could be by asking patients the to rate the *abstractness/concreteness* of a word framed into a *context-sentence* that should be mapped to a *word-sense* in WordNet. Assigning a proper *abstractness degree* to every *noun-sense* allows researching on *elastic* nouns, and on how a shift in countability affects the abstractness of a noun. Furthermore, it would be interesting to use the same methods used in this thesis on *elastic* nouns to test if the findings hold to truth for all nouns.

Bibliography

Alan C. Bale and David Barner. Semantic triggers, linguistic variation and the mass-count distinction. In *Count and Mass Across Languages*. Oxford University Press, 09 2012. ISBN 9780199654277. doi: 10.1093/acprof:oso/9780199654277.003.0013. URL `https://doi.org/10.1093/acprof:oso/9780199654277.003.0013`.

Marc Brysbaert, Amy Warriner, and Victor Kuperman. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46, 10 2013. doi: 10.3758/s13428-013-0403-5.

Chierchia. Mass nouns, vagueness, and semantic variation. *Synthese*, 174:99–149, 2010.

Jila Ghomeshi and Diane Massam. The count mass distinction: Issues and perspectives. In *Count and Mass Across Languages*. Oxford University Press, 09 2012. ISBN 9780199654277. doi: 10.1093/acprof:oso/9780199654277.003.0001. URL `https://doi.org/10.1093/acprof:oso/9780199654277.003.0001`.

N. Ide. The american national corpus: Then, now, and tomorrow. 2008.

Tibor Kiss, Francis Jeffry Pelletier, Halima Husic, Roman Nino Simunic, and Johanna Marie Poppek. A sense-based lexicon of count and mass expressions: The bochum English countability lexicon. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2810–2814, Portorož, Slovenia, may 2016. European Language Resources Association (ELRA). URL `https://aclanthology.org/L16-1447`.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Francis Jeffry Pelletier. Lexical nouns are both +mass and +count, but they are neither +mass nor +count. In *Count and Mass Across Languages*. Oxford University Press, 09 2012. ISBN 9780199654277. doi: 10.1093/acprof:oso/9780199654277.003.0002. URL https://doi.org/10.1093/acprof:oso/9780199654277.003.0002.

W. Gudrun Reijnierse, Christian Burgers, Marianna Bolognesi, and Tina Krennmayr. How polysemy affects concreteness ratings: The case of metaphor. *Cognitive Science*, 43(8):e12779, 2019. doi: https://doi.org/10.1111/cogs.12779. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/cogs.12779.

Roland Schaefer. Processing and querying large web corpora with the COW14 architecture. In Piotr BaÅ„ski, Hanno Biber, Evelyn Breiteneder, Marc Kupietz, Harald LÃ¼ngen, and Andreas Witt, editors, *Proceedings of Challenges in the Management of Large Corpora 3 (CMLC-3)*, Lancaster, 2015. UCREL, IDS. URL http://rolandschaefer.net/?p=749.

Tarun Tater, Diego Frassinelli, and Sabine Schulte im Walde. Concreteness vs. abstractness: A selectional preference perspective. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 92–98, Online, November 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.aacl-srw.13.

Robert X. Ware. *Some Bits and Pieces*, pages 15–29. Springer Netherlands, Dordrecht, 1979. ISBN 978-1-4020-4110-5. doi: 10.1007/978-1-4020-4110-5_2. URL https://doi.org/10.1007/978-1-4020-4110-5_2.

Martina Wiltschko. Decomposing the mass/count distinction: Evidence from languages that lack it. In *Count and Mass Across Languages*. Oxford University Press, 09 2012. ISBN 9780199654277. doi: 10.1093/acprof:oso/9780199654277.003.0009. URL https://doi.org/10.1093/acprof:oso/9780199654277.003.0009.

Roberto Zamparelli. Countability shifts and abstract nouns. In *Mass and Count in Linguistics, Philosophy, and Cognitive Science*, pages 191–224. John

Benjamins, 2020. URL https://www.jbe-platform.com/content/books/9789027260437-lfab.16.09zam.

A Models Evaluation

A.1 Evaluation Table of The Mass-Count Classifier (M_1)

The table of results that follows shows the performance of the 63 instances of the Mass-Count Classifier M_1 on different feature sets. The vector $\vec{V_i}$ corresponds to the features vectors in section 3.2. If a cell under $\vec{V_\text{i}}$ is marked with an 'X', the features present in $\vec{V_\text{i}}$ are utilized for the nouns' representation in the corresponding instance of the model. If a cell under $\vec{V_\text{i}}$ is marked with an '-', the features present in $\vec{V_\text{i}}$ are <u>not</u> utilized for the nouns' representation in the corresponding instance of the model.

Rank	$\vec{V_1}$	$\vec{V_2}$	$\vec{V_3}$	$\vec{V_4}$	$\vec{V_5}$	$\vec{V_6}$	Iterations scores	acc	std
1	X	X	-	X	X	X	0.98, 0.98, 0.95, 0.98, 0.96, 0.96, 0.96, 0.95, 0.97, 0.94	0.9621	0.0128
2	X	-	-	-	-	X	0.98, 0.95, 0.96, 0.98, 0.95, 0.98, 0.95, 0.94, 0.98, 0.95	0.9621	0.0149
3	X	-	-	-	X	X	0.98, 0.96, 0.95, 0.98, 0.95, 0.97, 0.95, 0.95, 0.98, 0.95	0.9613	0.012
4	X	X	-	-	X	X	0.98, 0.96, 0.95, 0.97, 0.97, 0.97, 0.95, 0.94, 0.98, 0.93	0.9597	0.0151
5	X	X	-	X	-	X	0.98, 0.97, 0.97, 0.97, 0.95, 0.97, 0.94, 0.92, 0.96, 0.96	0.959	0.0159
6	X	-	-	X	-	X	0.98, 0.94, 0.96, 0.98, 0.94, 0.95, 0.95, 0.95, 0.98, 0.95	0.9582	0.0155
7	X	-	-	X	X	X	0.99, 0.95, 0.95, 0.98, 0.95, 0.96, 0.95, 0.92, 0.97, 0.95	0.9582	0.0191

8	X	X	-	-	-	X	0.97, 0.96, 0.97, 0.97, 0.95, 0.98, 0.95, 0.94, 0.97, 0.93	0.9582	0.0148
9	-	X	-	X	X	X	0.99, 0.97, 0.95, 0.95, 0.95, 0.96, 0.94, 0.93, 0.97, 0.92	0.9535	0.0196
10	-	X	-	X	-	X	0.98, 0.97, 0.95, 0.96, 0.95, 0.95, 0.93, 0.92, 0.95, 0.95	0.9504	0.0156
11	-	X	-	-	-	X	0.98, 0.98, 0.94, 0.96, 0.95, 0.96, 0.92, 0.93, 0.94, 0.94	0.9496	0.0194
12	-	X	-	-	X	X	1., , 0.98, 0.94, 0.95, 0.94, 0.95, 0.93, 0.94, 0.94, 0.91	0.9481	0.0243
13	-	-	-	X	X	X	0.98, 0.96, 0.95, 0.96, 0.92, 0.96, 0.95, 0.94, 0.94, 0.91	0.9466	0.0182
14	-	-	-	X	-	X	0.98, 0.96, 0.95, 0.95, 0.94, 0.97, 0.95, 0.94, 0.93, 0.91	0.9458	0.019
15	X	X	-	-	X	-	0.95, 0.96, 0.92, 0.94, 0.97, 0.94, 0.95, 0.91, 0.96, 0.91	0.9412	0.02
16	-	-	-	-	-	X	0.98, 0.95, 0.92, 0.94, 0.91, 0.97, 0.93, 0.92, 0.93, 0.93	0.9388	0.0198
17	X	X	-	X	X	-	0.95, 0.95, 0.93, 0.95, 0.95, 0.95, 0.93, 0.92, 0.94, 0.91	0.9388	0.0133
18	-	-	-	-	X	X	0.96, 0.95, 0.95, 0.94, 0.91, 0.98, 0.93, 0.92, 0.93, 0.91	0.9365	0.0211
19	X	-	-	X	X	-	0.94, 0.94, 0.91, 0.95, 0.95, 0.93, 0.95, 0.89, 0.94, 0.91	0.9303	0.017
20	X	X	-	-	-	-	0.95, 0.93, 0.93, 0.91, 0.95, 0.93, 0.93, 0.91, 0.95, 0.91	0.9303	0.0144
21	X	X	-	X	-	-	0.94, 0.92, 0.91, 0.95, 0.95, 0.92, 0.93, 0.91, 0.95, 0.91	0.928	0.0147
22	X	-	-	-	X	-	0.94, 0.92, 0.91, 0.93, 0.94, 0.92, 0.91, 0.94, 0.95, 0.89	0.9257	0.0171
23	X	-	-	X	-	-	0.94, 0.93, 0.9, 0.95, 0.93, 0.92, 0.93, 0.91, 0.91, 0.9	0.9218	0.0154
24	X	-	-	-	-	-	0.94, 0.89, 0.92, 0.91, 0.9, 0.91, 0.91, 0.9, 0.92, 0.84	0.9063	0.0241
25	-	X	-	X	X	-	0.91, 0.93, 0.89, 0.91, 0.89, 0.91, 0.87, 0.86, 0.95, 0.87	0.8986	0.0266

26	X	X	X	-	X	X	0.91, 0.91, 0.87, 0.92, 0.87, 0.92, 0.91, 0.89, 0.89, 0.87	0.8955	0.0204
27	-	X	-	-	X	-	0.93, 0.92, 0.88, 0.91, 0.88, 0.88, 0.86, 0.84, 0.94, 0.88	0.8924	0.0312
28	X	X	X	X	-	X	0.9, 0.9, 0.87, 0.92, 0.92, 0.91, 0.88, 0.88, 0.87, 0.86	0.8909	0.0226
29	X	-	X	X	-	X	0.89, 0.92, 0.85, 0.91, 0.88, 0.91, 0.91, 0.89, 0.88, 0.87	0.8908	0.021
30	X	-	X	X	X	X	0.89, 0.93, 0.86, 0.91, 0.9, 0.91, 0.89, 0.85, 0.88, 0.85	0.8893	0.0258
31	-	X	-	X	-	-	0.92, 0.92, 0.89, 0.86, 0.88, 0.88, 0.84, 0.86, 0.9, 0.91	0.8869	0.0266
32	X	-	X	-	-	X	0.91, 0.92, 0.88, 0.89, 0.87, 0.9, 0.88, 0.87, 0.89, 0.85	0.8862	0.0198
33	-	X	X	X	-	X	0.9, 0.89, 0.87, 0.88, 0.88, 0.91, 0.9, 0.88, 0.86, 0.87	0.8854	0.0159
34	X	X	X	X	X	X	0.92, 0.92, 0.88, 0.88, 0.89, 0.91, 0.9, 0.86, 0.85, 0.86	0.8854	0.0223
35	-	-	X	-	-	X	0.9, 0.9, 0.84, 0.91, 0.89, 0.91, 0.9, 0.86, 0.87, 0.85	0.8839	0.0236
36	-	-	X	-	X	X	0.9, 0.88, 0.88, 0.88, 0.89, 0.91, 0.9, 0.88, 0.84, 0.85	0.8831	0.0201
37	X	X	X	-	-	X	0.9, 0.89, 0.86, 0.89, 0.91, 0.91, 0.86, 0.88, 0.85, 0.88	0.8831	0.0202
38	X	X	X	X	-	-	0.9, 0.89, 0.85, 0.91, 0.86, 0.91, 0.89, 0.87, 0.87, 0.84	0.88	0.0229
39	X	X	X	X	X	-	0.91, 0.88, 0.86, 0.89, 0.87, 0.89, 0.9, 0.88, 0.86, 0.84	0.8792	0.0189
40	X	-	X	-	X	X	0.91, 0.92, 0.85, 0.88, 0.87, 0.91, 0.9, 0.86, 0.86, 0.84	0.8792	0.0257
41	X	X	X	-	X	-	0.89, 0.91, 0.84, 0.91, 0.88, 0.91, 0.9, 0.88, 0.83, 0.84	0.8792	0.0279
42	-	X	X	-	X	X	0.87, 0.9, 0.86, 0.91, 0.86, 0.91, 0.89, 0.87, 0.88, 0.84	0.8785	0.0206
43	-	-	X	X	X	X	0.88, 0.89, 0.86, 0.89, 0.89, 0.91, 0.89, 0.88, 0.84, 0.85	0.8785	0.0206

44	-	X	X	X	X	X	0.9, 0.89, 0.85, 0.88, 0.89, 0.89, 0.89, 0.87, 0.85, 0.84	0.8754	0.0205
45	X	X	X	-	-	-	0.9, 0.91, 0.88, 0.88, 0.85, 0.88, 0.89, 0.86, 0.85, 0.84	0.8746	0.0204
46	-	-	-	X	X	-	0.85, 0.87, 0.91, 0.87, 0.86, 0.88, 0.88, 0.83, 0.91, 0.88	0.8739	0.0244
47	-	-	X	X	-	X	0.92, 0.88, 0.84, 0.88, 0.84, 0.91, 0.91, 0.87, 0.86, 0.83	0.8738	0.0299
48	-	X	X	-	X	-	0.88, 0.92, 0.85, 0.89, 0.87, 0.9, 0.87, 0.85, 0.84, 0.84	0.873	0.0247
49	-	X	X	-	-	X	0.89, 0.92, 0.83, 0.88, 0.84, 0.92, 0.89, 0.86, 0.86, 0.84	0.873	0.0304
50	-	X	X	X	X	-	0.86, 0.91, 0.84, 0.87, 0.88, 0.91, 0.9, 0.86, 0.85, 0.86	0.873	0.0228
51	X	-	X	X	X	-	0.88, 0.9, 0.85, 0.9, 0.87, 0.91, 0.87, 0.88, 0.85, 0.82	0.8723	0.0247
52	X	-	X	X	-	-	0.89, 0.88, 0.86, 0.88, 0.87, 0.89, 0.88, 0.86, 0.86, 0.84	0.8723	0.0169
53	X	-	X	-	-	-	0.88, 0.9, 0.85, 0.88, 0.88, 0.88, 0.9, 0.85, 0.85, 0.82	0.8707	0.0237
54	-	X	X	X	-	-	0.88, 0.89, 0.84, 0.88, 0.86, 0.9, 0.89, 0.87, 0.87, 0.84	0.8707	0.0203
55	-	X	-	-	-	-	0.92, 0.95, 0.87, 0.88, 0.86, 0.84, 0.83, 0.79, 0.91, 0.86	0.8699	0.0424
56	-	-	X	-	X	-	0.88, 0.89, 0.84, 0.88, 0.84, 0.9, 0.89, 0.87, 0.86, 0.82	0.8692	0.0243
57	-	-	X	-	-	-	0.89, 0.88, 0.87, 0.88, 0.87, 0.91, 0.88, 0.88, 0.83, 0.8	0.8684	0.0309
58	-	-	X	X	X	-	0.91, 0.89, 0.84, 0.88, 0.85, 0.88, 0.87, 0.85, 0.86, 0.83	0.8668	0.0224
59	-	X	X	-	-	-	0.88, 0.9, 0.81, 0.9, 0.88, 0.88, 0.88, 0.86, 0.85, 0.82	0.8668	0.0282
60	X	-	X	-	X	-	0.91, 0.89, 0.84, 0.88, 0.82, 0.9, 0.89, 0.84, 0.85, 0.83	0.866	0.0304
61	-	-	X	X	-	-	0.89, 0.88, 0.84, 0.85, 0.87, 0.89, 0.86, 0.87, 0.87, 0.81	0.8637	0.0219

62	-	-	-	X	-	-	0.85, 0.88, 0.86, 0.82, 0.86, 0.81, 0.84, 0.8, 0.88, 0.83	0.8428	0.0263
63	-	-	-	-	X	-	0.65, 0.62, 0.59, 0.69, 0.66, 0.64, 0.66, 0.71, 0.65, 0.63	0.6494	0.0337

A.2 Evaluation Table of The Abstractness Rate Predictor (M_2)

The table of results that follows shows the performance of the 63 instances of the Abstractness Rate Predictor M_2 on different feature sets. The vector $\vec{V_i}$ corresponds to the features vectors in section 3.2. If a cell under $\vec{V_\text{i}}$ is marked with an 'X', the features present in $\vec{V_\text{i}}$ are utilized for the nouns' representation in the corresponding instance of the model. If a cell under $\vec{V_\text{i}}$ is marked with an '-', the features present in $\vec{V_\text{i}}$ are <u>not</u> utilized for the nouns' representation in the corresponding instance of the model.

Rank	$\vec{V_1}$	$\vec{V_2}$	$\vec{V_3}$	$\vec{V_4}$	$\vec{V_5}$	$\vec{V_6}$	Iterations score	acc	std
1	-	X	X	X	X	X	0.5, 0.64, 0.57, 0.64, 0.62, 0.65, 0.61, 0.57, 0.7, 0.6	0.6107	0.0518
2	-	-	X	X	-	X	0.55, 0.62, 0.62, 0.59, 0.62, 0.62, 0.62, 0.58, 0.65, 0.63	0.6099	0.0282
3	X	X	X	-	X	X	0.55, 0.61, 0.61, 0.58, 0.67, 0.67, 0.57, 0.54, 0.63, 0.66	0.6092	0.0456
4	X	-	X	X	X	X	0.55, 0.6, 0.6, 0.62, 0.62, 0.6, 0.55, 0.6, 0.68, 0.6	0.6045	0.0347
5	-	X	X	-	X	-	0.56, 0.63, 0.64, 0.58, 0.62, 0.62, 0.6, 0.57, 0.65, 0.56	0.6037	0.0323
6	-	-	X	X	X	X	0.52, 0.58, 0.6, 0.59, 0.61, 0.65, 0.61, 0.57, 0.68, 0.6	0.6022	0.0411
7	-	-	X	-	-	X	0.47, 0.64, 0.6, 0.66, 0.66, 0.59, 0.57, 0.54, 0.66, 0.64	0.6022	0.059
8	-	X	X	-	X	X	0.55, 0.61, 0.6, 0.63, 0.64, 0.6, 0.57, 0.58, 0.64, 0.6	0.6014	0.0277
9	-	X	X	X	-	X	0.54, 0.6, 0.62, 0.6, 0.6, 0.62, 0.57, 0.57, 0.66, 0.62	0.5999	0.0322
10	-	-	X	-	X	-	0.53, 0.59, 0.63, 0.61, 0.63, 0.62, 0.58, 0.54, 0.7, 0.56	0.5991	0.0468

11	X	-	X	X	-	X	0.51, 0.6, 0.61, 0.63, 0.61, 0.65, 0.58, 0.57, 0.64, 0.59	0.5991	0.0397
12	-	X	X	-	-	-	0.55, 0.6, 0.57, 0.56, 0.61, 0.6, 0.6, 0.58, 0.68, 0.62	0.5968	0.0366
13	-	X	X	X	-	-	0.52, 0.61, 0.58, 0.59, 0.67, 0.59, 0.6, 0.55, 0.64, 0.63	0.5968	0.0416
14	-	-	X	X	X	-	0.51, 0.6, 0.58, 0.63, 0.64, 0.6, 0.6, 0.57, 0.64, 0.57	0.5953	0.0374
15	X	X	X	-	X	-	0.51, 0.62, 0.64, 0.6, 0.6, 0.61, 0.57, 0.53, 0.69, 0.6	0.5953	0.0496
16	X	X	X	-	-	X	0.49, 0.59, 0.59, 0.61, 0.63, 0.63, 0.58, 0.54, 0.67, 0.61	0.5945	0.0463
17	X	X	X	X	-	-	0.51, 0.58, 0.61, 0.62, 0.65, 0.59, 0.55, 0.61, 0.64, 0.57	0.5937	0.0416
18	X	-	X	X	X	-	0.54, 0.58, 0.58, 0.64, 0.58, 0.61, 0.56, 0.6, 0.7, 0.53	0.5937	0.0468
19	X	X	X	X	-	X	0.47, 0.58, 0.64, 0.6, 0.55, 0.67, 0.55, 0.56, 0.68, 0.63	0.593	0.0618
20	-	X	X	X	X	-	0.52, 0.56, 0.56, 0.64, 0.59, 0.61, 0.59, 0.59, 0.68, 0.57	0.5922	0.0429
21	X	X	X	X	X	X	0.53, 0.6, 0.64, 0.6, 0.6, 0.57, 0.54, 0.58, 0.65, 0.6	0.5914	0.0366
22	X	X	X	-	-	-	0.49, 0.62, 0.57, 0.62, 0.64, 0.6, 0.53, 0.58, 0.67, 0.57	0.5906	0.0499
23	X	X	X	X	X	-	0.51, 0.61, 0.63, 0.59, 0.63, 0.62, 0.54, 0.53, 0.67, 0.57	0.5898	0.0472
24	-	-	X	-	X	X	0.51, 0.58, 0.57, 0.66, 0.61, 0.6, 0.57, 0.53, 0.65, 0.61	0.5891	0.0453
25	-	-	X	X	-	-	0.51, 0.58, 0.57, 0.62, 0.6, 0.6, 0.57, 0.57, 0.67, 0.6	0.5883	0.0389
26	X	-	X	-	-	-	0.52, 0.57, 0.6, 0.6, 0.63, 0.59, 0.56, 0.55, 0.67, 0.59	0.5883	0.0419
27	X	-	X	-	-	X	0.52, 0.58, 0.6, 0.63, 0.57, 0.6, 0.54, 0.56, 0.67, 0.59	0.586	0.0407
28	X	-	X	-	X	X	0.52, 0.62, 0.6, 0.63, 0.59, 0.59, 0.56, 0.51, 0.65, 0.57	0.5836	0.0432

29	X	-	X	X	-	-	0.52, 0.6, 0.6, 0.62, 0.57, 0.6, 0.54, 0.58, 0.65, 0.54	0.5829	0.0369
30	-	X	X	-	-	X	0.48, 0.61, 0.57, 0.61, 0.65, 0.61, 0.57, 0.51, 0.69, 0.53	0.5829	0.0607
31	-	-	X	-	-	-	0.52, 0.59, 0.57, 0.59, 0.62, 0.59, 0.5, 0.57, 0.67, 0.57	0.579	0.0473
32	X	-	X	-	X	-	0.5, 0.55, 0.56, 0.6, 0.58, 0.6, 0.58, 0.57, 0.66, 0.57	0.5759	0.0382
33	X	X	-	X	-	X	0.46, 0.6, 0.56, 0.63, 0.57, 0.57, 0.55, 0.54, 0.57, 0.55	0.5596	0.0408
34	X	X	-	X	X	X	0.48, 0.57, 0.61, 0.61, 0.56, 0.58, 0.51, 0.53, 0.52, 0.51	0.5488	0.0432
35	-	X	-	-	-	X	0.44, 0.59, 0.57, 0.59, 0.57, 0.55, 0.52, 0.56, 0.56, 0.49	0.5434	0.0458
36	-	-	-	X	X	X	0.48, 0.53, 0.55, 0.53, 0.52, 0.57, 0.52, 0.59, 0.58, 0.53	0.5411	0.0309
37	-	X	-	X	-	X	0.47, 0.58, 0.57, 0.6, 0.54, 0.57, 0.53, 0.53, 0.5, 0.52	0.541	0.0386
38	-	X	-	-	X	X	0.47, 0.55, 0.55, 0.59, 0.59, 0.55, 0.5, 0.51, 0.55, 0.54	0.5403	0.0362
39	-	-	-	-	-	X	0.45, 0.56, 0.53, 0.61, 0.53, 0.53, 0.5, 0.57, 0.57, 0.53	0.5388	0.043
40	-	X	-	X	X	X	0.48, 0.56, 0.55, 0.57, 0.56, 0.6, 0.53, 0.53, 0.51, 0.5	0.5387	0.0349
41	X	X	-	-	-	X	0.46, 0.56, 0.55, 0.57, 0.55, 0.59, 0.5, 0.55, 0.55, 0.49	0.538	0.0381
42	-	-	-	X	-	X	0.45, 0.57, 0.54, 0.59, 0.5, 0.57, 0.52, 0.53, 0.54, 0.55	0.5372	0.0377
43	X	X	-	-	X	X	0.44, 0.54, 0.53, 0.57, 0.53, 0.59, 0.55, 0.57, 0.53, 0.52	0.5357	0.0384
44	-	-	-	-	X	X	0.46, 0.58, 0.53, 0.56, 0.52, 0.56, 0.48, 0.53, 0.53, 0.54	0.5294	0.0346
45	X	-	-	X	X	X	0.45, 0.54, 0.55, 0.59, 0.49, 0.54, 0.53, 0.53, 0.52, 0.53	0.5287	0.0344
46	X	-	-	X	-	X	0.45, 0.58, 0.52, 0.58, 0.5, 0.56, 0.48, 0.53, 0.5, 0.56	0.5263	0.04

47	X	-	-	-	-	X	0.46, 0.55, 0.53, 0.53, 0.53, 0.57, 0.5, 0.53, 0.54, 0.5	0.5248	0.0289
48	-	X	-	X	X	-	0.49, 0.55, 0.48, 0.56, 0.5, 0.56, 0.51, 0.48, 0.57, 0.53	0.5232	0.0337
49	X	X	-	X	X	-	0.46, 0.55, 0.53, 0.53, 0.5, 0.6, 0.58, 0.49, 0.53, 0.45	0.5217	0.045
50	X	-	-	-	X	X	0.44, 0.6, 0.5, 0.53, 0.51, 0.54, 0.52, 0.51, 0.54, 0.51	0.5217	0.0384
51	X	X	-	X	-	-	0.47, 0.55, 0.53, 0.57, 0.51, 0.57, 0.51, 0.5, 0.51, 0.48	0.5201	0.0316
52	-	X	-	X	-	-	0.47, 0.51, 0.51, 0.56, 0.53, 0.53, 0.52, 0.49, 0.51, 0.51	0.514	0.023
53	X	X	-	-	-	-	0.48, 0.52, 0.49, 0.54, 0.5, 0.53, 0.54, 0.49, 0.5, 0.49	0.5085	0.0231
54	X	X	-	-	X	-	0.45, 0.55, 0.53, 0.55, 0.45, 0.55, 0.56, 0.48, 0.5, 0.47	0.5085	0.0414
55	X	-	-	X	-	-	0.47, 0.5, 0.5, 0.5, 0.47, 0.53, 0.5, 0.51, 0.53, 0.51	0.5023	0.0197
56	-	X	-	-	X	-	0.42, 0.56, 0.51, 0.55, 0.47, 0.5, 0.51, 0.52, 0.5, 0.44	0.4977	0.0427
57	X	-	-	X	X	-	0.5, 0.51, 0.49, 0.5, 0.53, 0.5, 0.51, 0.51, 0.46, 0.46	0.4969	0.0229
58	-	-	-	X	-	-	0.46, 0.46, 0.49, 0.56, 0.48, 0.51, 0.47, 0.47, 0.58, 0.44	0.4915	0.0433
59	-	X	-	-	-	-	0.47, 0.52, 0.5, 0.53, 0.5, 0.5, 0.47, 0.47, 0.5, 0.45	0.4907	0.0233
60	-	-	-	X	X	-	0.45, 0.46, 0.45, 0.52, 0.47, 0.49, 0.47, 0.5, 0.48, 0.47	0.4753	0.0221
61	X	-	-	-	X	-	0.34, 0.32, 0.39, 0.31, 0.4, 0.39, 0.47, 0.4, 0.34, 0.4	0.3747	0.0467
62	X	-	-	-	-	-	0.28, 0.35, 0.38, 0.39, 0.38, 0.35, 0.4, 0.4, 0.3, 0.36	0.3584	0.0366
63	-	-	-	-	X	-	0.27, 0.28, 0.29, 0.32, 0.35, 0.29, 0.33, 0.36, 0.36, 0.37	0.3213	0.0361

A.3 Evaluation Table of The Abstract-Concrete Classifier (M_3)

The table of results that follows shows the performance of the 63 instances of the Abstract-Concrete Classifier M_3 on different feature sets. The vector $\vec{V_i}$ corresponds to the features vectors in section 3.2. If a cell under $\vec{V_\text{i}}$ is marked with an 'X', the features present in $\vec{V_\text{i}}$ are utilized for the nouns' representation in the corresponding instance of the model. If a cell under $\vec{V_\text{i}}$ is marked with an '-', the features present in $\vec{V_\text{i}}$ are <u>not</u> utilized for the nouns' representation in the corresponding instance of the model.

Rank	$\vec{V_1}$	$\vec{V_2}$	$\vec{V_3}$	$\vec{V_4}$	$\vec{V_5}$	$\vec{V_6}$	Iterations scores	acc	std
1	-	X	X	X	X	X	0.95, 0.96, 0.89, 0.92, 0.95, 0.91, 0.92, 0.94, 0.92, 0.95	0.9325	0.0211
2	-	X	X	-	X	-	0.92, 0.94, 0.88, 0.95, 0.95, 0.92, 0.96, 0.94, 0.91, 0.91	0.9288	0.0237
3	X	X	X	X	-	X	0.95, 0.94, 0.88, 0.92, 0.95, 0.91, 0.95, 0.92, 0.9 , 0.95	0.9275	0.0242
4	X	-	X	X	-	-	0.94, 0.94, 0.85, 0.91, 0.92, 0.92, 0.95, 0.96, 0.92, 0.94	0.9263	0.0288
5	-	-	X	-	-	X	0.92, 0.95, 0.86, 0.92, 0.95, 0.92, 0.95, 0.92, 0.92, 0.92	0.9263	0.024
6	X	-	X	-	X	X	0.95, 0.92, 0.89, 0.91, 0.96, 0.9 , 0.94, 0.92, 0.9 , 0.95	0.925	0.0237
7	-	X	X	-	-	-	0.95, 0.9 , 0.82, 0.94, 0.96, 0.94, 0.95, 0.94, 0.95, 0.9	0.925	0.0387
8	-	X	X	-	X	X	0.91, 0.91, 0.85, 0.94, 0.95, 0.91, 0.96, 0.95, 0.92, 0.94	0.925	0.0301
9	X	X	X	-	-	-	0.92, 0.91, 0.88, 0.91, 0.96, 0.91, 0.98, 0.94, 0.92, 0.91	0.925	0.0268
0	-	-	X	-	X	X	0.94, 0.94, 0.86, 0.92, 0.94, 0.91, 0.95, 0.91, 0.92, 0.95	0.925	0.0244

11	X	-	X	-	X	-	0.94, 0.91, 0.85, 0.95, 0.96, 0.92, 0.94, 0.91, 0.92, 0.94	0.925	0.029
12	X	X	X	X	X	X	0.94, 0.91, 0.88, 0.92, 0.96, 0.89, 0.95, 0.94, 0.92, 0.92	0.9238	0.0253
13	X	-	X	X	X	X	0.91, 0.91, 0.89, 0.92, 0.92, 0.91, 0.96, 0.94, 0.9 , 0.96	0.9238	0.0234
14	-	-	X	X	-	-	0.96, 0.94, 0.86, 0.92, 0.94, 0.91, 0.96, 0.92, 0.92, 0.88	0.9225	0.031
15	X	X	X	X	-	-	0.95, 0.91, 0.86, 0.92, 0.94, 0.91, 0.96, 0.94, 0.9 , 0.92	0.9225	0.0267
16	-	X	X	-	-	X	0.91, 0.91, 0.86, 0.92, 0.95, 0.92, 0.95, 0.92, 0.94, 0.92	0.9225	0.0236
17	X	-	X	-	-	X	0.9 , 0.9 , 0.88, 0.94, 0.96, 0.91, 0.96, 0.92, 0.91, 0.94	0.9225	0.0267
18	X	X	X	-	X	X	0.9 , 0.95, 0.89, 0.91, 0.92, 0.91, 0.95, 0.92, 0.92, 0.92	0.9212	0.0186
19	X	X	X	-	-	X	0.95, 0.91, 0.88, 0.95, 0.92, 0.91, 0.94, 0.92, 0.9 , 0.92	0.9212	0.0217
20	-	-	X	-	-	-	0.91, 0.94, 0.84, 0.92, 0.96, 0.92, 0.96, 0.9 , 0.91, 0.92	0.92	0.0336
21	X	X	X	X	X	-	0.92, 0.89, 0.88, 0.92, 0.96, 0.88, 0.96, 0.94, 0.92, 0.92	0.92	0.0302
22	-	-	X	X	-	X	0.92, 0.94, 0.86, 0.94, 0.94, 0.9 , 0.95, 0.92, 0.9 , 0.91	0.9188	0.0245
23	-	X	X	X	-	X	0.94, 0.9 , 0.85, 0.92, 0.94, 0.9 , 0.94, 0.94, 0.94, 0.92	0.9188	0.027
24	-	-	X	X	X	X	0.91, 0.92, 0.89, 0.91, 0.94, 0.9 , 0.95, 0.92, 0.9 , 0.94	0.9188	0.0188
25	X	X	X	-	X	-	0.91, 0.92, 0.88, 0.91, 0.95, 0.91, 0.95, 0.92, 0.9 , 0.92	0.9188	0.0211
26	-	X	X	X	-	-	0.91, 0.91, 0.85, 0.92, 0.94, 0.9 , 0.95, 0.92, 0.94, 0.94	0.9188	0.027
27	-	-	X	X	X	-	0.91, 0.92, 0.84, 0.92, 0.96, 0.9 , 0.92, 0.91, 0.95, 0.94	0.9188	0.0322
28	X	-	X	X	X	-	0.92, 0.92, 0.85, 0.92, 0.94, 0.9 , 0.95, 0.95, 0.91, 0.9	0.9175	0.0281

29	X	-	X	X	-	X	0.94, 0.9 , 0.88, 0.92, 0.94, 0.9 , 0.95, 0.91, 0.91, 0.92	0.9175	0.0211
30	X	-	X	-	-	-	0.92, 0.92, 0.84, 0.92, 0.96, 0.9 , 0.96, 0.9 , 0.91, 0.9	0.915	0.0339
31	-	X	X	X	X	-	0.9 , 0.88, 0.88, 0.91, 0.96, 0.9 , 0.96, 0.91, 0.9 , 0.9	0.91	0.0289
32	-	-	X	-	X	-	0.91, 0.9 , 0.86, 0.92, 0.91, 0.92, 0.94, 0.92, 0.91, 0.88	0.9088	0.0224
33	-	-	-	X	X	X	0.92, 0.88, 0.82, 0.89, 0.89, 0.81, 0.91, 0.88, 0.85, 0.88	0.8725	0.0334
34	-	-	-	X	-	X	0.92, 0.88, 0.82, 0.9 , 0.86, 0.82, 0.9 , 0.89, 0.85, 0.88	0.8725	0.031
35	-	X	-	X	-	X	0.89, 0.89, 0.85, 0.89, 0.9 , 0.82, 0.91, 0.86, 0.82, 0.89	0.8725	0.0289
36	-	-	-	-	-	X	0.9 , 0.85, 0.86, 0.86, 0.89, 0.84, 0.89, 0.89, 0.85, 0.9	0.8725	0.0215
37	X	X	-	-	-	X	0.84, 0.9 , 0.85, 0.89, 0.88, 0.82, 0.88, 0.88, 0.86, 0.91	0.87	0.0257
38	X	-	-	X	X	X	0.88, 0.9 , 0.88, 0.88, 0.91, 0.84, 0.85, 0.84, 0.82, 0.9	0.8688	0.0286
39	-	X	-	-	X	X	0.88, 0.88, 0.86, 0.86, 0.9 , 0.79, 0.9 , 0.89, 0.84, 0.89	0.8675	0.0322
40	X	-	-	-	-	X	0.89, 0.86, 0.86, 0.85, 0.88, 0.81, 0.91, 0.85, 0.88, 0.88	0.8662	0.025
41	X	X	-	X	-	X	0.84, 0.89, 0.8 , 0.88, 0.9 , 0.82, 0.89, 0.89, 0.85, 0.9	0.865	0.033
42	-	-	-	-	X	X	0.89, 0.86, 0.86, 0.85, 0.88, 0.82, 0.89, 0.88, 0.85, 0.88	0.865	0.0184
43	-	X	-	-	-	X	0.86, 0.85, 0.88, 0.86, 0.91, 0.81, 0.86, 0.89, 0.84, 0.89	0.865	0.0267
44	X	X	-	-	X	X	0.86, 0.88, 0.85, 0.86, 0.91, 0.81, 0.85, 0.88, 0.82, 0.91	0.8638	0.0308
45	X	-	-	-	X	X	0.89, 0.86, 0.86, 0.86, 0.86, 0.82, 0.88, 0.85, 0.84, 0.9	0.8625	0.0209
46	X	-	-	X	-	X	0.82, 0.88, 0.84, 0.88, 0.9 , 0.79, 0.88, 0.88, 0.86, 0.89	0.86	0.032

47	-	X	-	X	X	X	0.88, 0.88, 0.85, 0.86, 0.91, 0.81, 0.8 , 0.88, 0.85, 0.88	0.8588	0.0311
48	X	X	-	X	X	X	0.86, 0.86, 0.84, 0.88, 0.9 , 0.8 , 0.86, 0.88, 0.82, 0.86	0.8562	0.027
49	-	X	-	X	X	-	0.91, 0.86, 0.76, 0.81, 0.91, 0.81, 0.8 , 0.9 , 0.82, 0.81	0.8413	0.0497
50	X	X	-	X	-	-	0.86, 0.88, 0.84, 0.81, 0.82, 0.82, 0.8 , 0.89, 0.84, 0.8	0.8362	0.0287
51	X	-	-	X	-	-	0.84, 0.88, 0.82, 0.84, 0.81, 0.84, 0.84, 0.88, 0.8 , 0.8	0.8338	0.025
52	X	-	-	X	X	-	0.84, 0.9 , 0.79, 0.82, 0.82, 0.82, 0.81, 0.84, 0.82, 0.81	0.8288	0.0274
53	X	X	-	X	X	-	0.81, 0.89, 0.8 , 0.81, 0.86, 0.81, 0.79, 0.86, 0.78, 0.82	0.8238	0.0342
54	-	X	-	X	-	-	0.84, 0.81, 0.8 , 0.82, 0.86, 0.81, 0.75, 0.88, 0.82, 0.81	0.8212	0.0326
55	-	-	-	X	-	-	0.89, 0.79, 0.74, 0.8 , 0.82, 0.84, 0.82, 0.85, 0.78, 0.74	0.8062	0.0458
56	X	X	-	-	X	-	0.8 , 0.85, 0.78, 0.76, 0.81, 0.82, 0.74, 0.85, 0.79, 0.85	0.805	0.0376
57	X	X	-	-	-	-	0.79, 0.8 , 0.8 , 0.76, 0.79, 0.81, 0.74, 0.88, 0.81, 0.81	0.7988	0.0342
58	-	-	-	X	X	-	0.8 , 0.82, 0.75, 0.8 , 0.85, 0.75, 0.8 , 0.86, 0.75, 0.78	0.7962	0.0388
59	-	X	-	-	X	-	0.84, 0.8 , 0.75, 0.72, 0.84, 0.81, 0.7 , 0.88, 0.79, 0.78	0.79	0.0515
60	-	X	-	-	-	-	0.82, 0.76, 0.79, 0.72, 0.8 , 0.81, 0.68, 0.89, 0.78, 0.81	0.7862	0.0549
61	X	-	-	-	X	-	0.72, 0.76, 0.69, 0.7 , 0.68, 0.66, 0.71, 0.72, 0.69, 0.79	0.7125	0.0371
62	X	-	-	-	-	-	0.78, 0.76, 0.66, 0.61, 0.7 , 0.66, 0.74, 0.71, 0.65, 0.75	0.7025	0.0515
63	-	-	-	-	X	-	0.61, 0.65, 0.62, 0.59, 0.6 , 0.61, 0.68, 0.62, 0.61, 0.62	0.6225	0.0236