Study and Resolution of Arabic Lexical Ambiguity Through Transduction on Text Automaton

Nadia Ghezaiel[1(✉)] and Kais Haddar[2]

[1] Higher Institute of Computer and Communication Technologies of Hammam Sousse, Miracl Laboratory, Sousse, Tunisia
ghezaielnadia.ing@gmail.com
[2] Faculty of Sciences of Sfax, Miracl Laboratory, University of Sfax, Sfax, Tunisia
kais.haddar@yahoo.fr

Abstract. Lexical analysis can be a way to remove ambiguities in the Arabic language. So, their resolution is an important task in several domains of Natural Language Processing (NLP). In this context, this paper is inscribed. Our proposed resolution method is based essentially on the use of transducers on text automata. Indeed, these transducers specify the lexical rules of the Arabic language allowing corpus disambiguation. In order to achieve our resolution method, different types of lexical ambiguities are identified and studied. Then, an appropriate set of rules is proposed. After that, we represent all specified rules in NooJ. In addition, we present experimentation with NooJ platform conducted through various linguistic resources to obtain disambiguated syntactic structures suitable for the analysis. The results obtained are ambitious and can be improved by adding other rules and heuristics.

Keywords: Lexical ambiguity · Text annotation structure · Arabic lexical rule · NooJ transducer

1 Introduction

The need for disambiguation appears in several steps of analysis and applications such as syntactic analysis, recognition of named entities and morphological analysis. The disambiguation can be performed on different levels: morphological, syntactic and lexical levels. Indeed, disambiguating an Arabic corpus can widely facilitate several parsing processes which reduce largely the parsing time for researchers. For a successful resolution, we need a rigorous study of the Arabic language to facilitate the identification of rules which can be formalized through different frameworks. There are many theoretical platforms allowing formalization, such as grammars and finite state machines. In fact, finite automata and particularly transducers are increasingly used in NLP. Thanks to transducers, several local linguistic phenomena (e.g., recognition of named entities, morphological analysis) are treated appropriately. Transduction on text automata is so useful; it can remove paths representing morpho-syntactic ambiguities. Also, to formalize lexical rules, we need to find adequate criteria to classify lexical

T. Okrut et al. (Eds.): NooJ 2015, CCIS 607, pp. 123–133, 2016.
DOI: 10.1007/978-3-319-42471-2_11

rules in a specific order of application of rules and to define sufficient granularity levels of lexical categories allowing the identification of efficient rules. By these classifications, we aim to guarantee the optimization between rules, and to identify the disambiguation methods that can be exploitable by other steps of analysis.

In this context, our objectives are to study Arabic lexical ambiguities and to implement a lexical disambiguation tool for the Arabic language with NooJ platform through the transduction on text automaton. To do that, we need to identify and classify specific lexical rules for the Arabic language. Then, we implement these rules in NooJ platform and after that we call NooJ syntactic grammars in an adequate order to remove ambiguities existing in Text Annotation Structures (TAS).

In this paper, we begin by a state of the art presenting previous research interested in the resolution of ambiguities for the Arabic language. Next, we perform a study about lexical ambiguities. To resolve those lexical ambiguities, we establish transducers representing lexical rules. Then, we specify and test all these rules in NooJ linguistic platform.

2 State of the Art

Many studies aim to resolve Arabic lexical ambiguities at different levels: lexical, morphological, syntactic and semantic levels using different formalisms. In [12], the authors proposed a method for lexical disambiguation based on the cooperation between the morphological analyzer and the syntactic analyzer. In fact, all possible interpretations produced by the morphological analyzer will be the input of the syntactic one which consists in the application of constraints that are defined with the grammar rules. All grammar rules were specified in the Unification Based Grammar (UBG) formalism.

In [1], the author has developed a morphological syntactic analyzer for the Arabic language within LFG (Lexical Functional Grammar) formalism. The developed parser is based also on a cascade of finite state transducers for the sentence preprocessing and a set of syntactic rules specified in XLE (Xerox Linguistics Environment) for morphological analysis.

In [3], the proposed disambiguation method dealt with the 'alif-nûn' sequence in a given sentence. This method is based on the context-sensitive linguistic analysis to select the correct sense for a word in a given sentence without resorting to deep morpho-syntactic analysis.

Besides, in the last decades, we have witnessed a great increase in the number of systems which aim to disambiguate Modern Standard Arabic. Among those systems we mention MADA and TOKAN systems [7]. They are two complementary systems for Arabic morphological analysis and disambiguation process. Their applications include high-accuracy part-of-speech tagging, diacritization, lemmatization, disambiguation, stemming and glossing.

In [4], the system AMIRA is a set of tools built as a successor to the ASVMTools developed at Stanford University. The toolkit includes a tokenizer, a part of speech tagger (POS) and a Base Phrase Chucker (BPC). The technology AMIRA is based on

supervised learning with no explicit dependence on knowledge of deep morphology. This system treats partially the disambiguation process in Arabic.

Concerning the finite state tools, we find the linguistic environment Xerox [2] which is based on finite state technology tools (e.g. xfst, twolc, lexc,) for NLP. These tools are used in several linguistic applications such as morphological analysis, tokenization, and shallow parsing of a wide variety of natural languages. The finite state tools here are built on top of a software library that provides algorithms to create automata from regular expressions and equivalent formalisms and contains both classical operations, such as union and composition, and new algorithms such as replacement and local sequentialization.

Moreover, there are several works in NooJ platform that address the disambiguation process in Arabic. We cite the work presented in [6]. This work presents an approach of recognition and translation based on a representation model of Arabic Named Entities and a set of transducers resolving morphological and syntactic phenomena. We can also cite the work [13]. This work is based on HPSG formalism to identify all possible syntactic representations of the Arabic relative sentences. The authors explain the different forms of relative clauses and the interaction of relatives with other linguistic phenomena such as ellipsis and coordination. Besides, there are many works specialized in a particular phenomenon without taking into account other phenomena. We can cite the work described in [5] to analyze the Arabic broken plural. This work is based on a set of morphological grammars used for the detection of the broken plural in Arabic texts. In fact, those transducers are the basis for a tool generated by the linguistic platform.

As we can see, the Arabic disambiguation process is not yet performed because of some difficulties linked to the Arabic natural structure. Also, difficulties are linked to the lack of a perfect disambiguation tool for rule formalization. Moreover, previous NooJ works are limited to just one level or one linguistic phenomenon that reduces the rate of perfect disambiguation and decreases the reuse of some applications on account of their incompatibility and the absence of consistency between works and applications.

3 Arabic Lexical Ambiguity

In the following, we focus especially on three ambiguity generating areas in Arabic, which have the greatest impact in our work.

3.1 Unvocalization

Unvocalization can cause lexical ambiguities. Sentences of example (1) illustrate more this phenomenon.

(1) ذهب أحمد إلى المنزل كي يأخذ ذهب أمه إلى التاجر

Ahmad went to the house to take his mother's gold to the merchant

In example (1), the word ذهب can refer to the noun the gold in English, or the verb to go. Also, the word كتب can belong to several grammatical categories: verb or noun.

The meaning of this word will be very different depending on its class: if it is a plural noun, كتب means books, and if it is a verb, كتب means writing. Also, the word درب can refer to the name of a type of yellow fish or an existing road in the mountains or a verb to lead.

3.2 The Emphasis Sign (Shadda ّ)

In Arabic, the emphasis sign Shadda is equivalent to writing the same letter twice. The first letter would have 'Skoon' (ْ) and the second letter would have 'Fatha' (َ), 'Dhamma' (ُ) or 'Kasra' (ِ). For example, the word فَضَّلَ is actually فَضْضَلَ but, instead of writing 'ض' twice, we replace it with one 'ض' with Shadda on it. The insertion of Shadda changes the meaning of the word. For example, there is confusion between word دَرَسَ and word دَرَّسَ, because they have different meanings. دَرَسَ darasa means 'he studied' while دَرَّسَ darrasa means 'he taught'. Note that Shadda can be in the middle or at the end of the word.

The presence of Shaddah in the middle of the word can reduce some ambiguities linked to unvocalization. In fact, trough Shaddah we can identify the grammatical category of the word and easily attribute the right category. As an example, the word "قبل" is doubly ambiguous (Noun or verb) but, after the insertion of Shaddah at the middle of the word, the ambiguity decreases to one category (verb "قبّل").

3.3 Hamza

Hamza (همزة, hamzah) (ء) is a letter in the Arabic alphabet. It is not one of the 28 "full" letters, and the existence of this letter is due to historical inconsistencies in the standard writing system. Hamza is always written with its supports. They are three in number: the Alif أ Waw ؤ and Nabira ئ. The Hamza is written in different ways depending on its place in the word: at the begging, the middle or the end of the word.

The presence of Hamzas in all their types reduces the number of ambiguities and reduces the lexical category of the word. As an example, the word "اذن" can be doubly ambiguous (verb and noun) but, if we add the Hamza to this word, we decrease the number of ambiguities to just one lexical category.

3.4 Agglutination

In the Arabic language, particles, prepositions and pronouns can be attached to the adjectives, nouns, verbs and particles to which they relate. Compared to French, an Arabic word can sometimes match an English phrase. This characteristic generates a lexical ambiguity during the analysis. Indeed, it is not always easy to distinguish a proclitic or enclitic of an original character of the word. For example, the character "ف" in the word "فصل" (season) is an original character while in the word "فصل" (then he prayed), it is rather a proclitic.

3.5 Compound Words

Another common type of lexical ambiguity involves compound words in which we find two types of ambiguity. The first is linked to the meaning of words as shown in (2) and (3), respectively, and the second is linked to adjunction between words and interpretation of reading as shown in (4), (5) respectively.

(2) جلس الولد أمام الشّاشة الصّغيرة

The boy sat in front of television

(3) إنّ الشّاشة الصّغيرة في الحواسيب المحمولة ذات جودة عالية

Laptops have small screens with a high quality

In (2), "الشّاشة الصّغيرة" is a compound noun and, as such, a minimal unit for linguistic processing; therefore, the tag for الشّاشة الصّغيرة will have to contain relevant syntactic information, e.g. the form and type of complements of this unit. In (3), الشّاشة and الصّغيرة are distinct units that make up a free noun phrase.

(4) استعمل الحاسوب المحمول في العربة

I use a laptop in the vehicle

(5) استعمل الحاسوب المحمول في العربة

I use the computer which is portable in the vehicle

In examples (4) and (5), the compound word is related to the flexibility of reading and the comprehension of the sentence. In fact, we find in sentence (4) a strict reading in which the compound noun is "الحاسوب المحمول" (laptop) although in sentence (5) we find a flexible reading by taking the compound noun "المحمول في العربة" (which is portable in the vehicle).

4 Identification of Lexical Rules and Constraints

We carried out a linguistic study which allows us to identify 30 lexical rules resolving several forms of ambiguities. The identified rules were classified trough the mechanism of sub-categorization for verbs, nouns and particles.

4.1 Rules for Particles

Particles can be subdivided into three categories: particles acting on nouns, particles acting on verbs and particles acting on both nouns and verbs.

4.1.1 Particles Acting on Nouns

There are Arabic particles which must be followed by a noun like prepositions, particles of call, and particles of restriction. As an example, if we find prepositions like {مِنْ، إلى، عنْ، على، في، ب، لِ، ك، حتّى، رُبَّ، واو القسم، ت، حاشا، خلا،عدا}, then, they should be followed by a noun.

4.1.2 Particles Acting on Verbs

Particles can also be followed by a verb like subjunctive particles, apocopate particles, prohibition particles. As an example, if we find a subjunctive particle like {لن/ كي/ حتى/ لام التعليل/ إذن/ فاء السببية/ وأو المعية/ لام الجحود أن/ }, then, it should be followed by a verb.

4.1.3 Particles Acting on Both Nouns and Verbs

There are some particles that can be followed by a noun or a verb like particles of coordination or particles of explanation. To solve this ambiguity, we studied the context of the sentence, as an example of rules: if we find the particle of explanation "أن" then, it should be followed and preceded by a verb. Also, if we find a succession of two verbs, they should be separated by a particle like in the sentence "صلّى ثمّ نام" (he prayed then slept). The verb "صلى" (prayed) is succeeded by the particle "ثمّ" (then) then the verb "نام" (sleep). So, to solve an ambiguity linked to unvocalization, we can use the right and left context.

4.2 Rules for Verbs

We can apply the principle of sub-categorization to resolve the ambiguity linked to verbs. We are based essentially on the transitivity feature of verbs. In Arabic, a verb can be intransitive, transitive, double transitive and triple transitive. Either transitive or intransitive verbs can be transformed to transitive verbs with prepositions. Sentences of examples (6), (7) and (8) illustrate the Arabic transitivity mechanism.

(6) أكلت أختي وجبة لذيذة بسرعة (في مطبخنا) قبل ساعة

My sister ate a delicious meal quickly (in our kitchen) an hour ago

In example (6), the verb is "أكلت" (she ate) which is a transitive verb followed by a subject (noun) "أختي" (my sister), then an object (noun) "وجبة" (meal), and the remaining parts must be introduced through a particle like "في" which is a particle of preposition followed by a noun "مطبخنا" (our kitchen). Sentence (6) is composed of the verb (أكلت), the nominal phrase (أختي) (my sister), the nominal phrase (وجبة لذيذة), the prepositional phrase (بسرعة), the prepositional phrase (في مطبخنا), and the prepositional phrase (قبل ساعة).

(7) جلس أخي وحيدا في غرفته طوال اليوم

My brother sat alone in his room all day

In example (7), the verb "جلس" (he sat) is intransitive followed by a subject أخي (noun) (my brother), by an adverb (وحيدا) (alone) and two prepositional phrases. So, the sentence (7) is composed of the verb (جلس) (he sat), the nominal phrase (أخي) (my brother), by the adverb (وحيدا) (alone), by the prepositional phrase (في غرفته) (in his room) and by the prepositional phrase (طوال اليوم) (all day).

(8) خرج أخي من المكتبة منذ ساعتين

My brother came out from the library two hours ago

In example (8), the verb "خرج" (came out) is intransitive followed by a subject أخي (noun) (my brother), and two prepositional phrases. So, the sentence (8) is composed of the verb (خرج) (came out), the nominal phrase (أخي) (my brother), by the prepositional phrase (من المكتبة) (from the library) and by the prepositional phrase (منذ ساعتين) (two hours ago).

The mechanism of transitivity that is illustrated by the above sentences is summarized in the following table. Note that these examples respect the VSO order (Table 1).

Table 1. Transitivity summary table

Verb valency	Followed structures
Intransitive	NP (adverb) (PP)*
Transitive	NP NP (adverb) (PP)*
Double transitive	NP NP NP (adverb) (PP)*
Triple transitive	NP NP NP NP (adverb) (PP)*

4.3 Rules for Nouns

Concerning the sub-categorization of nouns, we are based on the contextual and lexical indices. In fact, the most reliable indicators for the detection and categorization are the right and the left contexts of a word. These indices are either internal or external contexts.

The internal indices are located inside the named entity. These are words that easily identify named entities. Example (9) illustrates the internal index.

(9) البنك العربي التّونسي
Arab Tunisian Bank

The word (البنك) (the bank) is an example of internal index.

The external index or right context refers to the context of an entity's occurrence in a sentence. In a speech, especially journalism, the author provides readers with additional information like people, places and organizations. This information can help to determine the type of an entity in an automatic process.

(10) مهدي جمعة الوزير الاول
Mehdi Jomaa, the Premier

The word (الوزير الأول) (the Premier) mentioned in example (10) shows an external index.

5 Proposed Approach

Now the formalization of extracted rules is finished and it is possible to apply our method of disambiguation. Our proposed approach consists of two main phases, the preprocessing phase and the application of the disambiguation process. The first phase consists in the segmentation, the agglutination of our corpus through morphological grammars and the annotation of the corpus through dictionaries. As an output of this phase, we get a TAS1 containing all possible annotations for the corpus. This TAS1 will be the input of the second phase. It contains all possible interpretations for an existing word in dictionaries. Note that the text annotation Structure of NooJ (TAS) is a representation type of text automaton.

The application of the disambiguation process consists in the suppression of wrong paths existing on the TAS1. This modification of TAS1 is the result of syntactic analyses which consist of the application of transducers representing lexical and contextual rules. These rules should respect a certain priority in their application from the most evident and intuitive rules until arriving at the least one (Fig. 1). The output of the disambiguation process will be a new TAS disambiguated containing the right paths and the right annotations.

The granularity is related to the existing lexical information in electronic dictionaries which may be more or less detailed, according to its nature and extension. The information in tags can be extended to 15 elements which make the information more detailed. This information has an important imprint in the disambiguation process. In fact, each level of lexical information can reduce the rate of ambiguous outputs. So, if the lexical information is detailed, the rate of granularity increases and the rate of ambiguous outputs decrease.

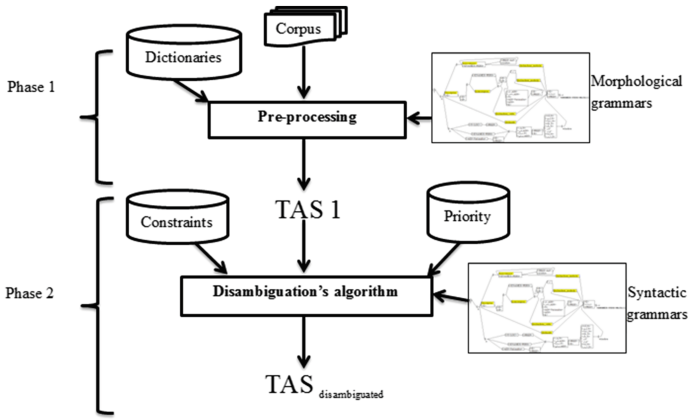

Fig. 1. Proposed approach

6 Implementation

The extracted rules have been formalized in the NooJ platform [11]. The process of disambiguation of text automaton is based on the set of the developed NooJ transducers. This set contains 17 grammars representing lexical and contextual rules. Figures 2 and 3 illustrate the implementation in NooJ of two lexical rules for Arabic particles.

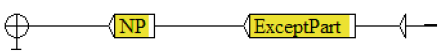

Fig. 2. Exception particle rules

The represented transducer of Fig. 2 indicates that if we recognize one of the specified exception particles, it must be followed by a noun phrase. The second transducer of Fig. 3 indicates that if we recognize one of the particles acting on verbs, then it must be followed by a verb.

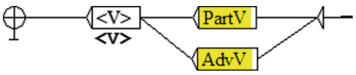

Fig. 3. Rule for particle acting on verbs

As we know, the Arabic sentence can be either verbal or nominal. So, we construct transducers to recognize these two specific forms. For a nominal sentence, it is generally formed by a topic and an attribute. Figure 4 indicates a transducer for recognition of a nominal sentence.

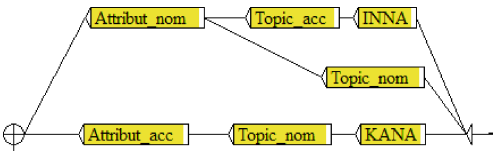

Fig. 4. Transducer representing a lexical rule for a nominal sentence

Figure 4 represents a transducer recognizing the nominal sentence; we distinguish different forms of topics and attributes. A nominal sentence can be formed by a nominative topic followed by a nominative attribute. Also, we can find the modal verb "KANA" followed by a nominative topic and an accusative attribute.

7 Experimentation and Evaluation

To experiment with our proposed method, we used a test corpus that contained 20000 meaningful sentences mainly from Tunisian newspapers and children's stories. Also, we used dictionaries containing 24732 nouns, 10375 verbs and 1234 particles. Besides, we used in our experimentation a list of morphological grammars containing 113 inflected verb form patterns, 10 broken plural patterns and 3 agglutination grammars. So, we used 17 graphs representing lexical rules, and a set of 10 constraints describing the execution of rules application. The obtained result is illustrated in Table 2.

Table 2 shows that 12000 sentences from the 20000 sentences existing in the corpus were totally disambiguated, which represents 60 %. Also, there are 6000 sentences partially disambiguated, which represents 30 %, and only 2000 sentences erroneously disambiguated, which represents 10 %. The partial disambiguation is due

Table 2. Table summarizing the obtained result

Corpus	Number	Percentage
Sentences	20000	100 %
Totally disambiguated	12000	60 %
Partial disambiguated	6000	30 %
Failed disambiguation	2000	10 %

to the lack of semantic rules. Also, sometimes, some rules were not correctly recognized. The erroneous disambiguation is linked to the lack of some information in our dictionaries which led to the wrong detection of left or right contexts.

During the disambiguation process, we got partially disambiguated sentences. This type of disambiguation is linked to different problems. These problems are due to the limited coverage of dictionaries as they did not contain all possible Arabic words. So, the produced TAS1 would be missing as well as the disambiguation process. Also, the lack of rules can be a source for partial disambiguating of sentences. In fact, the specificities of Arabic syntax may be the source of some additional processing difficulties. Besides, we need to elaborate other rules at different levels. There is another reason for the partial disambiguation which is linked to the granularity of lexical categories. Our evaluation is performed by the precision and recall measures (Table 3).

Table 3. Table summarizing the values of measures

Corpus	Precision	Recall	F-mesure
20000	0,6	0,9	0,72

In conclusion, the obtained results are ambitious and can be improved by adding other rules and heuristics. Thus, the creation of a tool allowing the transducer cascade generation that can be applied on the text automata is very useful. Such a tool can improve the obtained results.

8 Conclusion and Future Works

In this paper, we conducted a study on the different types of Arabic lexical ambiguities. This study allowed us to establish a set of lexical rules and constraints for lexical disambiguation. Established rules are specified with NooJ transducers. This disambiguation process will help us to reduce later parsing. Thus, an experiment is performed and satisfactory results are obtained. Also, we have shown the need to use the cascades on the text automata to simplify the ambiguity resolution process and make it more effective.

To perfectly annotate corpora, we need to enrich our resources by creating new dictionaries and new grammars representing the maximum of lexical rules. We need also to enrich our set of rules by adding new syntactic, morphological and semantic

levels and extend this methodology to other phenomena (i.e., coordination). As perspectives, we hope to continue our study of lexical disambiguation by writing new local grammars and also implementing a management module to build an automatic annotation tool for corpora. This module can be integrated later in the NooJ linguistic platform.

References

1. Attia, M.: Handlinh Arabic morphological and syntactic ambiguity within the LFG framework with a view to machine translation. Ph.D. thesis in the University of Manchester (2008)
2. Beesley, K.: Finite-state morphological analysis and generation of arabic at xerox research: status and plans. In: ACL/EACL, Toulouse, France, 6 July 2001
3. Dichy, J., Alrahabi, M.: Levée d'ambiguité par la methode d'exploration contextuelle: la sequence 'alif-nûn' en arabic. In: SIIE, Hammamet, Tunisia (2009)
4. Diab, M.: Second generation tools (AMIRA 2.0): fast and robust tokenization, POS tagging, and base phrase chunking. In: MEDAR 2nd International Conference on Arabic Language Resources and Tools, April, Cairo, Egypt (2009)
5. Ellouze, S., Haddar, K., Abdelhamid, A.: Etude et analyse du pluriel brisé arabe avec la plateforme NooJ. In: NooJ Conference and Workshop. Tozeur, Tunisia (2009)
6. Fehri, H., Haddar, K., Abdelhamid, A.: Recognition and translation of Arabic named entities with NooJ using a new representation model. In: FSMNLP, pp. 134–142 (2011)
7. Habash, N., Rambow, O., Roth, R.: MADA + TOKAN: a toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming ald lemmatization. In: Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt (2009)
8. Othman, E., Shaalan, K., Rafea, A.: Towards resolving ambiguity in understanding Arabic sentence. In: International Conference on Arabic Language Resources and Tools (2006)
9. Mesfar, S.: Morphological grammars for standard Arabic tokenization. In: Proceedings of the International NooJ Conference, pp. 114–127. Cambridge Scholars Publishing, Newcastle (2010)
10. Mesfar, S.: Analyse morpho-syntaxique automatique et reconnaissance des entités nommées en arabe strandard. Ph.D. thesis in the University of Franche Comté, 235 (2008)
11. Silberztein, M.: Disambiguation tools for NooJ. In: Proceedings of the 2008 International NooJ Conference, pp. 158–171. Cambridge Scholars Publishing, Newcastle (2010)
12. Shaalan, K., Othman, E., Rafea, A.: Towards resolving ambiguity in understanding Arabic sentence. In: The Proceedings of the International Conference on Arabic Language Resources and Tools, NEMLAR, Cairo, Egypt, 22–23 September 2004, pp. 118–122 (2004)
13. Zalila, I, Haddar, K.: Construction of an HPSG grammar for the Arabic relative sentences. In: The Proceedings of RANLP, Hissar, Bulgaria (2011)