Computing Semantic Textual Similarity based on Partial Textual Entailment

Martin Víta
NLP Centre, Faculty of Informatics, Botanická 68a, 602 00 Brno, Czech Republic

1 INTRODUCTION

Nowadays, textual entailment is a well-founded notion. There are several definitions of textual entailment: for our purposes, we will use the following one (Androutsopoulos and Malakasiotis, 2010):

"By textual entailment is understood a relationship between coherent text T and a language expression H, which is considered as a hypothesis. T *entails* H if the meaning of H as interpreted in context of T, can be deduced from the meaning of T."

Recognizing textual entailment (abbreviated often as RTE) is a decision problem whether T entails H. During the last ten years textual entailment attracts an intensive attention of the NLP community. RTE is currently a deeply studied problem having consequences for many different applications of NLP – including multi-document summarization, machine translation evaluation, student response analysis, etc.

RTE is closely related to a problem of paraphrases recognizing. A paraphrase s' of a sentence s is a sentence that has the same or *almost the same* meaning as s in a given context. The relationship between paraphrasing and textual entailment is straightforward: a paraphrase can be considered as a mutual textual entailment (s entails s' and s' entails s simultaneously). Due to this fact methods for RTE and paraphrasing are often treated together, although some differences are taken into the account.

RTE is a binary decision problem. When obtaining a negative result, it is not possible to say if T almost entails H, i. e. how close is H to a sentence that is entailed by T. Roughly said, textual entailment is crisp and rigid. The notion of partial textual entailment is an attempt to incorporate situations such that T "partially" entails H. According to (Levy et al., 2013), we say that an ordered pair (T,H) forms a partial textual entailment if *a fragment* of the hypothesis H is entailed by T.

To obtain a better idea of the problem, we provide an illustration on three examples:

1. Wonderworks Ltd. constructed the new bridge.
2. The new bridge was constructed by Wonderworks Ltd.
3. Wonderworks Ltd. constructed the new bridge over the river Thames.
4. A new bridge over the river Thames was constructed.

The first two sentences are mutual paraphrases. The first (or the second one) and the third one form a partial textual entailment. The last one is entailed by the third one.

The main aim of the doctoral project is to investigate new methods for recognizing partial textual entailment in both mono- and cross-lingual setting, describe methods for computing semantic textual similarity based on a partial textual entailment score, develop a system for recognizing partial textual entailment and implement functionality of computing a semantic textual similarity within a real-world application framework.

2 MOTIVATION

Although the topic is probably interesting from the theoretical point of view, the motivation for the proposed project arises mainly from practical issues currently being solved in the author's practice (regarding R&D policy and management).

- *Recommendation of similar documents in R&D domain* (i. e. papers, patents, ...) related to a given project (containing similar ideas) for a reviewer in order to provide an information support.
- *Discovery of (potentially) duplicate projects* - projects with similar content, i. e. with proposals with a big amount of sentences that are mutual (partial) paraphrases. The goal is straightforward – to avoid financing of the same or closely related thing twice from the public sources.
- *Identification of groups of R&D results of a single author based on a repeatedly used idea* – for example, papers describing an application of a cer-

Víta, M..
Computing Semantic Textual Similarity based on Partial Textual Entailment.
In *Doctoral Consortium (DC3K 2015)*, pages 3-12

tain method on slightly different objects of research or reusing ideas presented in conference papers later in journal papers, publishing slightly changed versions of papers in different languages, etc. This phenomenon is typical for systems were "funding depends on quantity" and it leads to distortions in measuring the real performance of an individual or an institution. We will refer this issue as "multiple reporting task".

Another particular application of our system will be in the field of in-depth exploring (medical) curricula in order to improve author's results concerning creating a balanced content of medical study. Parts of the study – courses, disciplines, learning units etc. – are represented as textual information in plaintext files. (As we will mention in the further text, the language of these representations has some important features.) When exploring a given curricula or comparing courses of different faculties/universities, it is important to answer questions like "if the content of a given course is (partially) covered by another course", "what is the similarity of two courses" and identify parts of overlapping contents. Current approaches are based on a simple bag-of-words representation and a cosine similarity (in some cases improved by LSA application). These approaches are not able to capture valuable aspects such as paraphrasing of parts of course descriptions.

According to the potentially practical utilization, there are some limitations and requirements of the final application:

- maximally reduced usage of external tools, especially NLP tools (in except lemmatization and/or stemming),
- language independence whenever possible.

The proposed approach should also enable further extending towards cross-lingual setting (that allows us to compute a semantic textual similarity of documents in different languages).

As mentioned in (Nevěřilová, 2014b), "the typical attribute of the current state-of-the-art in this area [textual entailment] is that a number of articles describe methods (with possible applications) whereas few articles describe applications of the proposed methods in large systems. If successful, the proposed thesis can fill this gab and turn methods into a real-world application.

The hypothesis is that the approach proposed in this paper based on the (partial) textual entailment is significantly better than LSA based methods (both mono- and multilingual).

3 STATE-OF-THE-ART

This section provides a brief overview of state-of-the-art: it describes the keystones of our approach – recognizing textual entailment, recognizing partial textual entailment and word2vec model. It also regards semantic textual similarity and plagiarism detection.

3.1 Recognizing Textual Entailment

One of the definitions of the notion of *textual entailment* was provided in the previous chapter. The textual entailment differs from other kinds of entailment (such as logical or analytical entailment) – the description of distinctions among these types of entailment is out of scope of this work since it is application-oriented. The relevant discussion concerning this topic is summarized in (Nevěřilová, 2014b).

For reader's convenience, we are going to introduce a commonly used notation: $T \rightarrow H$ will be an abbreviation of "T entails H" or equivalently "H is entailed by T. The other case when $T \rightarrow H$ does not hold, we will use $H \not\rightarrow T$. As mentioned above, recognizing textual entailment task is a binary decision task whether $H \rightarrow T$ or $H \not\rightarrow T$. In the further text we will also write that an ordered pair (T,H) is a textual entailment whenever $T \rightarrow H$.

As mentioned in the previous section, the definition of the textual entailment is strict (in a sense of an arbitrary, but widely accepted definition): if $T \not\rightarrow H$, there is no way how to measure how close is H to some H' such that $T \rightarrow H'$. In other words, from the RTE viewpoint, a hypothesis H completely unrelated to the text T is treated in a same way as a hypothesis H' that is "almost entailed".

Classification of RTE Approaches

An up-to-date comprehensive classification of RTE approaches is provided in (Nevěřilová, 2014b). This classification arises from the classification introduced in an older survey (Androutsopoulos and Malakasiotis, 2010), but enriches it by adding a higher level of classification – methods are divided to basic and advanced methods.

The basic approaches are characterized by *dealing with sequences of words.* To this class we assign: methods based on:

- B-o-W (bag-of-words) approaches: these methods are based on surface string similarity, in some cases after certain preprocessing is applied. The main idea is to match words in H with "most suitable" words in T. Several string similarity mea-

sures are used (e. g. edit distance). These approaches are usually straightforward in case of paraphrasing detection since sentences involved have approximately the same length.

- Vector space approaches: these methods deal with vector representation of T and H and computing their similarity (often by the cosine distance). This approach was successfully used on paraphrasing task by (Erk and Padó, 2009).

As mentioned in (Nevěřilová, 2014b), the main advantage of these approaches is their relative language independence. Their main disadvantage is the inability to handle expressions that do not preserve the truth value, e. g. negations if expressed as separate words.[1]

In contrast, advanced approaches *deal with the structure of the text.*

To this class belong the following methods:

- Logic-based approaches: the core of these approaches is a mapping of T and H to logical expressions Φ_T and Φ_H (for each possible reading of T, H, respectively) and checking the logical entailment, usually in form $(\Phi_T \wedge B) \models \Phi_H$, where B stands for corresponding logical representation of common knowledge. This part of the task is done using a theorem prover. Logical formalisms being taken into the account, contain mainly first order logic capturing even temporal aspects (Tatu and Moldovan, 2006) and description logics (de Salvo Braz et al., 2006). The crucial question of these approaches is obtaining the common knowledge. Typical knowledge bases used as starting point are WordNet (Fellbaum, 1998), Extended WordNet (Moldovan and Rus, 2001), FrameNet and VerbNet.
- Syntactic similarity approaches: these methods deal with (dependency) tree representations and use more or less sophisticated computations ranging from simple common edge count (Malakasiotis and Androutsopoulos, 2007) to tree edit distance (Kouylekov and Magnini, 2005), sometimes combined with lexical sources like in (Kouylekov and Magnini, 2006). Other approaches compare the parse tree of H with subtrees of T (Zanzotto et al., 2009).
- Approaches based on similarity measures over symbolic meaning representations: in this case, semantic representation of H is compared with a semantic representation of T. Again, FrameNet or WordNet knowledge bases are used.
- Approaches based on decoding: idea of these approaches is the application of transformation rules like replacing synonyms, hyponyms/hypernyms replacements, paraphrase patterns etc. Such transformations can be associated with confidence scores learned from the corpus and (T,H) is decided to be a textual entailment in case of the sum of maximum-score sequence is greater or equal than a given threshold. This approach – combined with probabilistic methods – was used in (Harmeling, 2009).

Approaches based on machine learning methods have a separate category in (Androutsopoulos and Malakasiotis, 2010) that cannot be simply transferred to this hierarchy since their background varies from using simple surface strings features to advanced features derived from semantic representations.

For completeness let us mention that RTE is only one of several tasks connected with textual entailment. Other ones are namely textual entailment generation and textual entailment extraction. Indeed, these tasks are not relevant from our perspective.

Selection of Existing Systems, Test Suites and Corpora

Although the textual entailment tasks are widely studied at least in the last five years, the number of RTE systems is relatively low.

A respectable source of existing functional RTE systems is the ACL Web Wiki page. In the period of writing this proposal, six functional systems were presented, namely: VENSES (based on two subsystems: a reduced version of GETARUN which produces the semantics from complete linguistic representations and a partial robust constituency-based parser), Nutcracker (a system using first order logic – a theorem prover and a finite model builder), EDITS – Edit Distance Textual Entailment Suite, BIUTEE – Bar-Ilan University Textual Entailment Engine, formerly separate application, now a part of the EOP, based on dealing with dependency trees and performing knowledge-based transformations (Stern and Dagan, 2012), EXCITEMENT Open Platform (EOP - a generic architecture and a comprehensive implementation for textual inference in multiple languages. The platform includes state-of-art algorithms, a large number of knowledge resources, and facilities for experimenting and testing innovative approaches

[1] From our point of view there arises a natural question whether these language constructions occur in "scientific papers" or texts describing curricula, such as syllabi (that are both important for our purposes) less or more often than in corpora where experiments with RTE were performed. It is expected that several collections of texts involved in our work will not contain this kind of expressions – but this is a hypothesis that should be proved.

(**?**) and TIFMO, a system based on Dependency-based Compositional Semantics (DCS) and logical inference (Tian et al., 2014).

The BIUTEE system will be recalled in the next section.

Since RTE become popular NLP task, there appear several test suites or corpora in order to compare results of different systems. Well known collection of RTE test suites were prepared for RTE workshops. During 2004-2013, eight of these workshops took place – first as Pascal RTE Challenges then as tracks of Text Analysis Conference and the last as a track of SemEval challenge. Links to these datasets are provided within ACL Wiki, some of them are available for direct download, some are freely available upon a request. We also should point out two other corpora: Microsoft Research Paraphrase Cospus (MSR) and Boeing-Princeton-ISI (BPI). The first one is – after (Nevěřilová, 2014a) – most widely used benchmark for paraphrase recognition. It contains more than 5000 sentences from which more than half is annotated as paraphrases. The second one is focused on textual entailment. Compared with Pascal RTE suites, according to (Clark, 2006), BPI is simpler in terms of syntax but more challenging in the semantic viewpoint, with the intention of focusing more on the knowledge rather than just linguistic requirements. Other corpora are again listed in ACL Wiki in Textual Entailment Resource Pool section.

For our purposes, the most interesting collection of test suites comes from The Joint Student Response Analysis and 8th Recognizing Textual Entailment Challenge at SemEval-2013 Task 7. They were inspired by developments of tutorial dialogue systems (Dzikovska et al., 2012). It contains the SciEntsBank corpus (Dzikovska et al., 2013), that was originally developed to assess student answers in a very fine-grained level. Moreover, SciEntsBank Extra (Nielsen et al., 2008) contains additional annotations that break down answers into "facets" or low-level concepts and relationships connecting them.

3.2 Partial Textual Entailment

The core of our work is a development of a system for recognizing partial textual entailment. In this section we introduce the notion of a partial textual entailment and a related notion of a faceted textual entailment. Then, we describe one existing system – approach to recognizing partial textual entailment that will serve us as a starting point for our research.

The Notion of Partial Textual Entailment and Its Motivation

The fragments of an idea of partial textual entailment were introduced in (Nielsen et al., 2009), although this notion was not explicitly mentioned in the paper. Partial textual entailment began to be elaborated in recent years by Omer Levy (Levy et al., 2013). It is a "response" to above-mentioned rigidity of textual entailment.

Following up the paper (Levy et al., 2013), let us consider a couple of sentences:

- $T :=$ *Muscles generate movement in the body.*
- $H :=$ *The main job of muscles is to move bones.*

Obviously, T does not entail H. Nevertheless we "feel there is some relationship between T and H", such that H is *almost* entailed by T. Thus, it is reasonable to ask how close is T and H to entailment. So there arises a need for a graded approach.

Recall the previously mentioned definition, we say that an ordered pair (T,H) forms a partial textual entailment if a fragment of the hypothesis H is entailed by T.

To distinguish these both forms of textual entailment, for previously defined defined notion we will use the expression *complete textual entailment*. Trivially, if (T,H) forms a complete textual entailment, then (T,H) forms a partial textual entailment – converse generally does not hold, i.e. the condition that each fragment of H is entailed by T does not necessarily ensure the complete textual entailment, when the fragments have bounded length (for example: expressing of sequences of actions where ordering is important.)

In (Nielsen et al., 2009) Nielsen at al. have defined the notion of a *facet* in this setting. Given a hypothesis H, a facet is an ordered pair of words (w_1,w_2) that are both contained in H accompanied with the direct semantic relation between w_1,w_2. In (Levy et al., 2013), they use a simplified model when facet is considered as the pair of words w_1,w_2 without an explicit expression of the semantic relation. This simplified model will be also suitable for our purposes due to certain characteristics of word2vec model. We will recall this remark in the following section.

Now we are able to state a definition of recognizing faceted entailment. Recognizing faceted entailment is a binary classification task whether the facet (w_1,w_2) contained in the hypothesis H is expressed or unaddressed by the text T (Levy et al., 2013).

Returning to our example, the facet *(muscles, move)* refers to the agent role in H and is expressed in the text T, whereas *(move, bones)* not.

Obviously, the faceted entailment – in this simplified version omitting the semantic relation between parts of the facet – is a partial textual entailment.

System for Recognizing Faceted Entailment and Its Modules

In (Levy et al., 2013) it is also proposed a system for recognizing faceted entailment. This system will be also a starting point for our improvements.

It consists of three independent modules such that each one for given inputs – a text T and a facet (w_1, w_2) – returns the result of recognizing faceted entailment:

- *Exact Match* – T is represented as a B-o-W containing all lemmas and tokens from T. If both lemmas of w_1 and w_2 are contained in this B-o-W, than the decision is positive, otherwise not.

  Such exact match was used as a baseline in several textual entailment challenges (Bentivogli et al., 2011).

- *Lexical Inference* – By this module it is checked whether both words w_1 and w_2 *or semantically related words* appear in T. Similarity score of given words is computed using Resnik similarity measure (Resnik, 1995) over WordNet (Fellbaum, 1998). If the value of similarity score between w_i and a word t_j from T is greater or equal to a given threshold (authors empirically set this threshold to 0.9), than the match of w_i and t_j is accepted. If both words from the facet have their matches in T, then the decision is positive, otherwise not.

- *Syntactic Inference* – This module is based on the previously mentioned BIUTEE system. It operates on dependency trees and applies a sequence of knowledge-based transformations converting T to H. The entailment is determined depending on the "cost" of generating hypothesis from the text.

  The BIUTEE deals with dependency trees, thus both T and the given facet must be parsed. To obtain the dependency tree of the facet, the following steps are processed: parsing H (obtaining dependency tree of H), locating nodes referring to w_1 and w_2, finding their lowest common ancestor a within the H's dependency tree and selecting a path from w_1 to w_2 via a. This path is a dependency tree of the facet and it is transferred along with the dependency tree of T to BIUTEE as inputs. The BIUTEE result is taken as decision of recognizing faceted entailment involved.

This system was examined in different configurations (employing different combinations of its three modules) over the SciEntsBank corpus where the facet decomposition was already made, i. e. corresponding facets were provided – they were automatically extracted from the corpus and manually selected/checked.

These configurations were:

1. *Baseline: ExactM*
2. *BaseLex: ExactM* $\vee$ *LexicalI*
3. *BaseSyn: ExactM* $\vee$ *SyntacticI*
4. *Disjunction: ExactM* $\vee$ *LexicalI* $\vee$ *SyntacticI*
5. *Majority: ExactM* $\vee$ *(LexicalI* $\wedge$ *SyntacticI)*

In different scenarios (i. e. different subsets of the corpus), the Majority configuration outperforms all other configurations – using F_1 measure it achieved results from 0.765 to 0.816 (depending on the scenario), since the Baseline result varies from 0.670 to 0.713 and BaseLex from 0.710 to 0.760. The complete table of results is provided in (Levy et al., 2013).

As shown by the authors (Levy et al., 2013), a system for recognizing partial textual entailment can be used even for RTE. The process contains three consequent steps:

1. Decompose the hypothesis into facets.
2. Determine whether each facet is entailed.
3. Aggregate the individual facet results and decide on complete textual entailment accordingly.

Since the authors used already prepared facets, the first step was obtained "free of charge" (facets were prepared as a part of training/testing data). When building a system for RTE based on recognizing partial textual entailment, an auxilary application for facet decomposition (implementing the first step) has a big influence on the overall result, i. e. wrong decomposition may lead to inferior results.

Current Systems based on Partial Textual Entailment

The idea of partial textual entailment and/or faceted entailment is so far used in a few systems – focused mainly on student response analysis or grading (Burrows et al., 2015). There are also intentions to use this concept in text summarization (Gupta et al., 2014) or attempts to use it when processing tweets (Rudrapal and Bhattacharya, 2014).

3.3 Related Issues: Semantic Text Similarity and Plagiarism Detection

Both above-mentioned notions of textual entailment and partial textual entailment are related to the problem of *semantic textual similarity*. We will follow

the meaning of this notion presented in (Agirre et al., 2015):

"Given two snippets of text, semantic textual similarity (STS) captures the notion that some texts are more similar than others, measuring their degree of semantic equivalence. Textual similarity can range from complete unrelatedness to exact semantic equivalence, and a graded similarity score intuitively captures the notion of intermediate shades of similarity, as pairs of text may differ from some minor nuanced aspects of meaning to relatively important semantic differences, to sharing only some details, or to simply unrelated in meaning."

STS and textual entailment differ in several properties: STS is a bidirectional graded equivalence of text snippets (Agirre et al., 2015), whereas textual entailment deals with "direction" and this notion is not graded. Indeed, partial textual entailment can serve as a starting point for establishing the STS relation between text snippets as we propose. Similarly as textual entailment, STS attracts attention of NLP community due to wide range of potential applications, containing among others plagiarism detection, dialogue systems etc. These topics are probably the inspiration of furthercoming SemEval-2016 Task 1 challenge.

Plagiarism detection task is often considered as a possible application of textual entailment. In Merriam-Webster-Online-Dictionary, the meaning of plagiarism is:

- to steal and pass off (the ideas or words of another) as ones own
- to use (anothers production) without crediting the source
- to commit literary theft
- to present as new and original an idea or product derived from an existing source.

Obviously, discovering plagiarism of certain types is in principle equivalent to solving our issues mentioned in the Introduction, namely the "duplicity task" and "multiple reporting". Hence, methods we are going to investigate, are potentially useful in plagiarism detection.

Let us notice that the first and/or last bullet also cover translating existing works and the last one also contains "self-plagiarism". As mentioned in (Rehurek, 2008), plagiarism is an act of crime. Hence, it is a conscious act. In R&D funding, the assumption of consciousness is not so important – it is also necessary to detect similar projects independently proposed by different institutions. Indeed, (our) way of solving this issue rely only on the semantic content not on the background.

The borderline between a fair treatment and a plagiarism is not crisp – plagiarism is "a fuzzy notion": as previously mentioned, turning conference contributions to regular journal papers is more-or-less acceptable practice.

Employing scoring based on a partial textual entailment when computing STS and a consequent application in a real-world system for duplicity/plagiarism detection was not investigated yet.

3.4 Keystone of a Furthercoming Approach: word2vec Model

Word embeddings are low-dimensional vector representations of words. Nowadays, *word2vec* model belongs to the most popular word embedding model – according to number of practical applications used in various semantic tasks including machine translation or sentiment analysis.

Word2vec model arises from the idea of predicting the neighbours of a word using a neural network. The (vector) representations of words are learned using the distributed Skip-gram or Continuous Bag-of-Words (CBOW), (Mikolov et al., 2013a). The CBOW idea is to predict the word "in the middle" from the surrounding words, whereas in Skip-gram model the training objective is to learn vector representations that are good at predicting its context in the same sentence. Because of its simplicity, the Skip-gram and CBOW models can be trained on a large amount of text data: in a parallelized implementation (`code.google.com/p/word2vec`) can learn a model from billions of words in hours (Mikolov et al., 2013b).

Word2vec model belong to a class of distributed representations for words. The main attribute of distributed representations (proposed relatively long time ago, in the second half of 80th in (Williams and Hinton, 1986)), is that the representations of (semantically) similar words are close in the vector space.

Word2vec representations capture many linguistic regularities and many types of similarities that can be expressed as linear translations, (Mikolov et al., 2013c). As an illustration we provide a well known example: $\mathit{representation}(\text{king}) - \mathit{representation}(\text{man}) + \mathit{representation}(\text{women})$ is close to $\mathit{representation}(\text{queen})$.

The vectors represent relationships between concepts via linear operations. For example, vector $\mathit{representation}(\text{France}) - \mathit{representation}(\text{Paris})$ is close to the vector $\mathit{representation}(\text{Italy}) - \mathit{representation}(\text{Rome})$, (Mikolov et al., 2013b).

This model has a solid mathematical/computer science background, we are going to use some of the

characteristics of this model "from the user's view", omitting formalization of optimization tasks being solved when learning the neural network.

Word2vec provides two basic tools to use with these vector representations: *distance* and *analogy*. The distance tool returns a list of the closest neighbours of a given word w.r.t. cosine similarity over the vector representation. The analogy tool allows us to query for regularities captured in the vector model through simple vector subtraction and addition, (Miñarro-Giménez et al., 2015).

Word2vec model has already been employed in solving semantic similarity tasks (but in the different manner as proposed here), for example in WHUHJP system for estimating similarity of tweets (Xu et al., 2015), word2vec representations were also used as features for ML approaches to recognizing textual entailment (Bjerva et al., 2014).

We expect that in the final version of Ph.D. thesis, a comparison of systems for semantic similarity presented in SemEval will be presented.

Employing word2vec Model in a Cross-lingual Environment

Word2vec model can be successfully used in a bilingual environment in terms of generating and extending dictionaries and phrase tables (Mikolov et al., 2013b). The basic idea is relatively simple having very little assumptions about the languages involved: missing (unknown) translations are obtained by learning language structures over large monolingual data and mapping between languages on a small domain (in terms of the mapping). In other words, having a set of concepts/notions, their word representations have a similar geometric arrangements in both vector spaces (corresponding with source and target language).The authors achieve almost 90 % precision@5 for translation between English and Spanish, more in (Mikolov et al., 2013b) where also several visual, self-explanatory representations are provided.

More formally, let us have n word pairs and their vector representation $(x_i, z_i)_{i=1}^{n}$, where $x \in R^{d_1}$ is a vector representation of i-th word in the source language and $z \in R^{d_2}$ a vector representation of its translation. The goal is to find a matrix W such that Wx_i approximates z_i. The matrix W is obtained as a solution of an optimization problem:

$$\min_{W} \sum_{i=1}^{n} \|Wx_i - z_i\|^2.$$

In (Mikolov et al., 2013b) it is solved with stochastic gradient descent. When a translation of a new word is needed, then we take its vector representation x in source language space and compute $z = Wx$. The last step is to find out the word such that its representation (in the target vector space) is the closest to z in sense of cosine similarity.

4 CURRENT WORK – BASELINE APPROACH

For recommending similar documents and duplicity task as well as exploring medical curricula, a solution based on latent semantic analysis (LSA) is being currently tested.

The overall principle is simple: documents that are taken into the account are transformed into a plaintext form, then a document-term-matrix (DTM) with tf-idf weighting is created. Dimensionality of this DTM is consequently reduced by LSA. For similarity computations cosine distance is used.

Pairs of documents where cosine similarity is greater or equal to a given threshold are returned as potentially duplicate. In recommendation task, top n most similar documents are obtained for a given document. Results of these "traditional" approaches will serve as a baseline for our experiments further with duplicates and recommending similar documents.

5 AIM OF THE DOCTORAL PROJECT

In this section we are going to break the proposed project into a set of consequent issues and describe their main ideas.

5.1 Main Issues – Plan of the Work

The first issue is an investigation of new methods for recognizing partial textual entailment. The starting point will be the architecture of the previously mentioned system (Levy et al., 2013). The idea is to modify the Lexical Inference module in the sense of replacing the former calculation of word similarity based on WordNet by dealing with distances obtained from word2vec model. No knowledge technologies (like WordNet) will be used. This task also requires (training and testing) data sets to be prepared. This issue will be finished by the comparison of original Levy's system and the new one.

The second issue is extending the architecture towards cross-lingual setting. The task is to decide whether the text T in the source language entails a facet (w_1, w_2) in the target language. The proposed

method arises from the Lexical Inference module, again within the word2vec framework: having a word w_1 in the target language – assuming we already have computed the liner mapping φ (represented by the matrix W – see the previous chapter). For w_1 we calculate $\varphi^{-1}(w_i)$ and compute its cosine similarity to representation t_j of a word from the hypothesis. We will consider such t_j from the hypothesis, such that cosine similarity is the lowest. If the similarity is higher than a given threshold and for w_2 so, then T and (w_1, w_2) constitutes a (cross-lingual) partial textual entailment, otherwise not. This issue will be analogously as the previous one completed by a comparison – in this case, we will compare results of the monolingual variant and the bilingual.

The third issue is an automatic extraction of facets. In both issues above, it was assumed (similarly as in (Levy et al., 2013)) that facets are obtained externally/manually from the hypothesis. Thus, a natural question arises: whether (and how) this process can be automated by ML methods. At first, we would like to check if it is possible to use only the features derived from the word2vec representation or it is necessary to employ features that come from the syntactic parsing of the hypothesis. This issue requires proposing of suitable evaluations of the process of generating facets from a given hypothesis.

The fourth issue is the development of the scoring method for STS task. Having two sentences S and T, we can compute the percentage of facets from T that are entailed by S. Given two text snippets A and B and a sentence S from A, we are able to find out a sentence T from B such that this percentage is the highest among all sentences in B. We obtain an entailment score with respect to (A,B) by averaging this percentages over all sentences in A. In a similar way we can define a paraphrase score with respect to (A,B). Therefore we obtain a quadruple of values:

1. an entailment score with respect to (A,B)
2. an entailment score with respect to (B,A)
3. a paraphrase score with respect to (A,B)
4. a paraphrase score with respect to (B,A)

This quadruple of values models the "entailment"-relationship between documents. For instance, if A is a summarization of B, then the first and third value will be greater or equal to the second and fourth value. The particular goal is to investigate how to turn these scores (using probably also other features of A and B) into a single value that will be used for estimating semantic textual similarity of a pair of documents and also implement a web service that will implement these computations.

The fifth issue is a development of a decoding module. As shown in (Miñarro-Giménez et al., 2015), word2vec can be used for capturing different semantic relations, e.g. hypernym/hyponym, membership, etc. The main idea of this issue is to replace Syntactic inference module of the system described in the previous chapter by analogous module that will not deal with dependency trees but with sequences of vectors. According to the basic classification, this approach belongs to decoding methods. Operations such as replacing a word by its hypernym will be processed as replacing the corresponding vector representation in order to transform the initial text T representation to an ordered pair of vectors close to the vector representation of considered facet.

The sixth issue is a development and evaluation of an application for "duplicate task" and "recommendation task". These applications will have a form of web services. The evaluation will be performed using the same metrics traditionally employed in the evaluation of STS systems (particularly SemEval-2016 Task 1), i. e., mean Pearson correlation between the system output and the gold standard annotations. The goal is to present a system that will provide better results in recommending semantically similar documents than currently developed system that uses LSA.

It should be adjusted that the primary goal of this work is to create a real-word application(s)/components solving mentioned issues rather than achieving good results in standardized benchmarks. Non-excellent results on standardized tasks can be balanced by the simplicity of the entire system and consequent easy maintenance. Nevertheless, we expect that results of proposed R(P)TE system in standard evaluations will be comparable with other up-today recognizing (partial) textual entailment systems.

6 CONCLUSION

In this work, we recall and discuss the notion of (partial and faceted) textual entailment and we propose a system for recognizing partial textual entailment based on the word2vec model. We present an idea how to extend this proposed system for recognizing partial textual entailment to multilingual environment. The aim of the doctoral project is to use this system to calculate STS based on scores derived from partial textual entailment features (among multilingual documents). We want to achieve better results than standard methods based on LSA.

REFERENCES

Agirre, E., Banea, C., et al. (2015). Semeval-2015 task 2: Semantic textual similarity, english, s-panish and pilot on interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), June*.

Androutsopoulos, I. and Malakasiotis, P. (2010). A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, pages 135–187.

Bentivogli, L., Clark, P., Dagan, I., Dang, H., and Giampiccolo, D. (2011). The seventh pascal recognizing textual entailment challenge. *Proceedings of TAC*, 2011.

Bjerva, J., Bos, J., van der Goot, R., and Nissim, M. (2014). The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity. *SemEval 2014*, page 642.

Burrows, S., Gurevych, I., and Stein, B. (2015). The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25(1):60–117.

Clark, Fellbaum, H. (2006). The Boeing-Princeton-ISI (BPI) textual entailment test suite. http://www.cs.utexas.edu/ pclark/bpi-test-suite/.

de Salvo Braz, R., Girju, R., Punyakanok, V., Roth, D., and Sammons, M. (2006). An inference model for semantic entailment in natural language. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 261–286. Springer.

Dzikovska, M. O., Nielsen, R. D., and Brew, C. (2012). Towards effective tutorial feedback for explanation questions: A dataset and baselines. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 200–210. Association for Computational Linguistics.

Dzikovska, M. O., Nielsen, R. D., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., and Dang, H. T. (2013). Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. Technical report, DTIC Document.

Erk, K. and Padó, S. (2009). Paraphrase assessment in structured vector space: Exploring parameters and datasets. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 57–65. Association for Computational Linguistics.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Gupta, A., Kaur, M., Singh, A., Goel, A., and Mirkin, S. (2014). Text summarization through entailment-based minimum vertex cover. *Lexical and Computational Semantics (* SEM 2014)*, page 75.

Harmeling, S. (2009). Inferring textual entailment with a probabilistically sound calculus. *Natural Language Engineering*, 15(04):459–477.

Kouylekov, M. and Magnini, B. (2005). Recognizing textual entailment with tree edit distance. In *Proceedings of the PASCAL RTE Challenge*, pages 17–20.

Kouylekov, M. and Magnini, B. (2006). Combining lexical resources with tree edit distance for recognizing textual entailment. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 217–230. Springer.

Levy, O., Zesch, T., Dagan, I., and Gurevych, I. (2013). Recognizing partial textual entailment. In *ACL (2)*, pages 451–455.

Malakasiotis, P. and Androutsopoulos, I. (2007). Learning textual entailment using svms and string similarity measures. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 42–47. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751.

Miñarro-Giménez, J. A., Marín-Alonso, O., and Samwald, M. (2015). Applying deep learning techniques on medical corpora from the world wide web: a prototypical system and evaluation. *arXiv preprint arXiv:1502.03682*.

Moldovan, D. I. and Rus, V. (2001). Logic form transformation of wordnet and its applicability to question answering. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 402–409. Association for Computational Linguistics.

Nevěřilová, Z. (2014a). Paraphrase and textual entailment generation. In *Text, Speech and Dialogue*, pages 293–300. Springer.

Nevěřilová, Z. (2014b). *Paraphrase and Textual Entailment Generation in Czech [online]*. PhD thesis, Faculty of Informatics, Masaryk University Brno.

Nielsen, R. D., Ward, W., and Martin, J. H. (2009). Recognizing entailment in intelligent tutoring systems. *Natural Language Engineering*, 15(04):479–501.

Nielsen, R. D., Ward, W., Martin, J. H., and Palmer, M. (2008). Annotating students understanding of science concepts. In *In Proc. LREC*.

Rehurek, R. (2008). *Semantic-based plagiarism detection [online]*. Ph.d. thesis proposal, Faculty of Informatics, Masaryk University Brno.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*.

Rudrapal, D. and Bhattacharya, B. (2014). Recognition of partial textual entailment for bengali tweets. *Social-India 2014*, 2014:29.

Stern, A. and Dagan, I. (2012). Biutee: A modular open-source system for recognizing textual entailment. In *Proceedings of the ACL 2012 System Demonstrations*, pages 73–78. Association for Computational Linguistics.

Tatu, M. and Moldovan, D. (2006). A logic-based semantic approach to recognizing textual entailment. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 819–826. Association for Computational Linguistics.

Tian, R., Miyao, Y., and Matsuzaki, T. (2014). Logical inference on dependency-based compositional semantics. In *Proceedings of ACL*, pages 79–89.

Williams, D. R. G. H. R. and Hinton, G. (1986). Learning representations by back-propagating errors. *Nature*, pages 523–533.

Xu, W., Callison-Burch, C., and Dolan, W. B. (2015). Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval)*.

Zanzotto, F., Pennacchiotti, M., and Moschitti, A. (2009). A machine learning approach to textual entailment recognition. *Natural Language Engineering*, 15(04):551–582.