1B2-OS-25b-4

意味と表記の組み合わせによる用例ベースの質問応答モデル

Example-based QA model with combining semantic and surface similarity

川端 貴幸 *1　　佐藤 一誠 *2
Takayuki Kawabata　　Issei Sato

*1株式会社サイバーエージェント　　*2東京大学
Cyberagent Inc.　　Tokyo University

In this research, we aim to improve the accuracy of response choices for example-based dialog modeling. We propose a model combining semantic similarity and surface similarity to overcome paraphrase by synonyms or incompleteness due to erroneous input. We show on a evaluation set used for the task of FAQ reply that our method outperforms baseline methods that work under the same condition.

1. はじめに

近年, 企業とユーザの新しいコミュニケーションとして, チャット上での接客が増加している. チャットは, メールと電話の中間に位置するコミュニケーションツールであり, 若い世代を中心にコミュニケーション手段の中心となりつつある. また, 自然言語処理の分野へも RNN や LSTM など深層学習の応用 [Vinyals 15] が進んだことにより, 雑談を行うチャットボットなどが盛り上がりをみせている. このような背景の元, チャット上でチャットボットによる接客を行う実サービスが急増している. 例えば, アスクルは, 同社が運営する EC サイトの問い合わせにおいてチャットボットを導入することで, 省人化効果は 6.5 人分となったと試算している. また, チャットボットの導入により, オペレーターの活時間が創出され, より丁寧なサポートが求めれる問い合わせの対応にオペレーターが注力できるようになり, 顧客満足度の向上にも繋がっているという.

このような簡単な問い合わせに対して自動的に回答を返すシステムとしては，用例ベースの対話システム (example-based dialog modeling:EBDM)[Lee 09][Lasguido 16] が一般的である. 用例ベースの対話システムは，発話と応答の組である用例を用いて，ユーザが入力した発話に対して用例の中から適切な応答を選択して返す対話システムである. この枠組みでは，用例データベース（以下，用例 DB と呼ぶ）の品質と，用例 DB からの応答選択の精度という二つの要素が用例ベース対話システムの品質の決定に大きな影響を与える.

本研究では，用例 DB からの応答選択についての精度向上を目指す. 我々の手法では，意味上の類似度と表記上の類似度を組み合わせたモデルにより, 同義語による言い換えや, 誤入力による不完全文に対してもロバストな応答選択法を提案する. 比較のベンチマークとしては IBM Watson の日本語 Natural Language Classifier(NLC)*1 とし，提案手法の優位性を検証する.

2. 用例ベースの質問応答システム

雑談など一般的な非タスク型の対話システムでは, ユーザ発話 q とそれに対するシステム応答 r のペア $< q, r > \in D$ の集合から用例 DB は構成される. しかし，企業への問い合わせに対して適切な応答が求められるようなタスク型の場合には，システム応答 r_i に対して，表 1 のように言い換え可能な複数のユーザからの想定質問 q_{ij} のペア $< q_{ij}, r_i > \in D$ で構成されることが一般的である.

連絡先: 川端貴幸，株式会社サイバーエージェント，kawabata_takayuki@cyberagent.co.jp

*1 http://www.ibm.com/smarterplanet/jp/ja/ibmwatson/developercloud/nl-classifier.html

用例ベースの質問応答は，ユーザの質問 q' が与えられた時に，用例 DB から最も適切な応答 r_i を探す問題である.

$$r' = \arg\max_{r_i \in D} S(q', r_i) \tag{1}$$

多クラス分類の問題と定義して，q' から直接 r_i のスコアを求めるアプローチと，q' と q_{ij} の類似度を求め，最も類似している q_{ij} を持つ用例 $< q_{ij}, r_i >$ の r_i を求めるアプローチに大別される.

$$r' = \arg\max_{<q_{ij}, r_i> \in D} sim(q', q_{ij}) \tag{2}$$

上の式 (2) の $sim(q', q_{ij})$ は 2 つの文の類似度を計算するシンプルなモデルであり，対話システムのアプリケーションを設計する上で再利用性の高いモデルである点などを考慮して我々は後者のアプローチを選択する.

表 1: FAQ 応答の用例 DB の例

質問		応答	
パスワードが分からない	q_{11}	パスワードが分からない場合は …	r_1
パスワードを忘れた	q_{12}	パスワードが分からない場合は …	r_1
pass が分からない	q_{13}	パスワードが分からない場合は …	r_1
ID が分からない	q_{21}	ID が分からない場合は …	r_2

3. 関連研究

2 つの文の類似度を計算する方法として，単純な方法は表記上の近さを計算する方法である. 例えば，編集距離や Longest Common Subsequence を用いたものや，bag-of-ngrams, bag-of-words などで文字の一致率をみる方法がある，また，情報検索で伝統的に使われる手法として,bag-of-words に TF・IDF で重み付けしたベクトル間のコサイン類似度を用いる方法もある. これらの表記上の近さを計算する方法は，同義語など意味的に同じだが表記が異なる単語に対して明らかに弱い.

それに対して，意味上の類似度を考慮する方法が活発に研究されている. 最近では，word2vec[Mikolov 13] や GloVe[Pennington 14] などの単語の分散表現を用いて，文ベクトルを構成し文ベクトル間のコサイン類似度を計算する方法や,CNN を用いて複数の特徴量を抽出しタスクに応じて予測する方法 [Zhang 15] などが主流となっている. また，機械翻訳を NN のみで実現する encoder-decoder アーキテクチャの拡張として, 文の分散表現を GRU などで合成し, その分散表現から周辺の文を予測するような decoder を学習する Skip-Thoughts モデル

[Kiros 15] が提案されており, 文の類似性判定で高い性能を報告している. しかし, これら NN を用いた手法は大量のコーパスと計算リソースが性能向上には必要とされる.

4. 提案手法

4.1 2つの文の類似度を予測するモデル

2つの文の類似度を予測するために，教師ありの機械学習を用いる.Algorithm1 に学習フェーズでの予測モデルを生成する疑似コードを載せる. 学習データは，文のペアとその類似度を示すラベルの集合である. 事前に学習済みの単語の分散表現辞書と，単語の重み辞書を用いることで，文のペアから複数の意味上の特徴量を抽出する. また，同様に文のペアから複数の表記上の特徴量も抽出する. それぞれについての特徴量は後述する. すべての文のペアに対して上記特徴量を計算し作成した特徴量行列と，類似度のラベルから，2つの文の類似度を予測するモデルを教師付きで学習する. なお，モデルは類似するかしないかの 2 値を予測する分類器でもよいし，連続値である類似度を予測する回帰でもよい.

Algorithm 1 文の類似度を予測するモデル生成の疑似コード

Input: 文ペアのリスト $[(s_{11}, s_{12}), (s_{21}, s_{22}), \cdots, (s_{n1}, s_{n2})]$
Input: 類似度のラベルのリスト $L = [l_1, l_2, \cdots, l_n]$
Require: 単語の分散表現辞書 WE
Require: 単語の重み辞書 WW
Require: 意味上の特徴抽出器のリスト $[sfe_1, sfe_2, \cdots, sfe_m]$
Require: 表記上の特徴抽出器のリスト $[lfe_1, lfe_2, \cdots, lfe_h]$
Output: 学習済みの文の類似度予測モデル M

```
F ⇐ 空の特徴量行列
for i = 1 to n do
    f⃗ ⇐ []
    for j = 1 to m do
        f⃗ ⇐ concat(f⃗, sfe_j((s_i1, s_i2), WE, WW))
    end for
    for k = 1 to h do
        f⃗ ⇐ concat(f⃗, lfe_k((s_i1, s_i2)))
    end for
    F[i] ⇐ f⃗
end for
M ⇐ train_model(F, L)
```

4.2 意味上の特徴量

4.2.1 文の分散表現行列

意味上の特徴量として，単語の分散表現に基づいた文の分散表現を求め，そのコサイン類似度を用いる. 単語集合 V 中の各単語は d 次元の実数値ベクトルで表現される. 単語の分散表現として d 次元の実数値ベクトルを計算するためには word2vec を用いた. すべての単語は $L \in \mathbb{R}^{d \times |V|}$ の行列として構成される. ここで $|V|$ は語彙数である. 文として n 個の単語系列が与えられたとき，i 番目の単語の分散表現は L から一致する単語 v_i を検索し，$x_{v_i} \in \mathbb{R}^d$ となる. 文の中のすべてのワードベクトルは以下の行列として表される.

$$X = (x_{v_1}, x_{v_2}, \cdots, x_{v_n}) \tag{3}$$

4.2.2 畳み込み

文の分散表現として固定次元のベクトルを得るために [Zhang 15] を参考にいくつかの畳み込み操作を用いる.X の中の各列 r に対して以下のように average, min, max, argmax_abs の 4 つの畳み込み操作を行うことで, 各次元ごとにそれぞれ異なる特徴量が得られる.

$$c_r^{agv} = \frac{1}{n}\sum_{i=1}^{n} X_{r,i} \tag{4}$$

$$c_r^{min} = min(X_{r,1}, X_{r,2}, \cdots, X_{r,n}) \tag{5}$$

$$c_r^{max} = max(X_{r,1}, X_{r,2}, \cdots, X_{r,n}) \tag{6}$$

$$c_r^{argmax\_abs} = \underset{|X_{r,i}| \in X_r}{\arg\max} X_{r,i} \tag{7}$$

各次元ごとに上記の畳み込み操作を行うことにより行列 X から d 次元のベクトルが得られる. それぞれの畳み込みベクトルを，C^{avg},C^{min},C^{max},C^{argmax_abs} とする.

4.2.3 分散表現のノルムと重みづけについて

単語 v_i に対応する要素が 1 でそれ以外 0 であるような $|V|$ 次元の指標ベクトル e_{v_i} を導入する. e_{v_i} は, 異なる単語同士の内積を 0 とした最も単純な分散表現とみることができる.

文書の表現で一般的な単語頻度ベクトルは, 単語 v_i の文書中での頻度を $|v_i|$ と表記すると，$|v_i|e_{v_i}$ の畳み込みとみることができる．すなわち, 文書の頻度表現は単語の指標ベクトル e_{v_i} の $|v_i|$ による重みづけされた畳み込みと考えることができる．また, 文書分類などのタスクでは頻度ベクトルではなく，IDF などを考慮した単語 v_i に対応した量 $I(v_i)$ を用いることで性能が向上することが知られているが，これは $|v_i|e_{v_i}$ を $I(v_i)e_{v_i}$ へと変更したことに対応する．また,e_{v_i} はノルムが 1 の単位ベクトルであるのでこの変更はノルムの変更に対応する.

分散表現 x_{v_i} を $\|x_{v_i}\|\frac{x_{v_i}}{\|x_{v_i}\|}$ と分解して考え，x_{v_i} は単位ベクトル $\frac{x_{v_i}}{\|x_{v_i}\|}$ に対してノルム $\|x_{v_i}\|$ によって重みづけされた表現とみなすことができる．つまり, 単に分散表現を文書中で畳み込んだ場合，各単語の重みは $\|x_{v_i}\|$ に依存することとなる.

頻度ベクトル表現の場合と同様に, 分散表現に関しても，その重みづけに対応するノルム $\|x_{v_i}\|$ を $I(v_i)$ で置き換えることで同様の効果があることが期待できる. 本研究では，用例 DB($< q_{ij}, r_i > \in D$) の質問文中に含まれる単語 t の重み I(t) として IDF に類似した下記を用いる.

$$I(t) = log\frac{|r|}{rf(t) + smooth} \tag{8}$$

$|r|$ は用例 DB 中に含まれる応答 r_i の種類数であり,rf(t) は単語 t を含む質問 q_{ij} と紐づく応答 r_i の種類数である.

したがって，式 3 において,$\|x_{v_i}\|\frac{x_{v_i}}{\|x_{v_i}\|}$ を $I(v_i)\frac{x_{v_i}}{\|x_{v_i}\|}$ へとノルムを修正したベクトルを用いる.

分散表現を用いて単語間関係の類推推論を行う場合, 主にコサイン類似度が用いられるが，これはノルムを 1 に正規化した分散表現を用いていることになる．すなわち, ノルムを 1 に正規化した分散表現は単語間の類似度を計算するのに有用であることを示唆している．また, 先に説明した通りノルムは他の単語との畳み込みを考慮する上では, その単語の重要度（重みづけ）という意味をもつ．元々の分散表現のノルムは分散表現の学習コーパスに依存した値になっているため, 他のタスクや異なるコーパスへ適応する場合には, 修正が必要な可能性がある．$I(v_i)$ によるノルムの修正は, このような状況を考慮した手法である.

4.2.4 未知語の扱い

今回，word2vec の学習コーパスとしては日本語 wikipedia のコーパスを用いて，最小出現頻度のハイパーパラメータを 40 として学習した. 学習された word2vec の単語集合 V に含まれない単語は未知語と呼ばれ，文ベクトルを計算する際の未知語の扱いについては自明ではない. 未知語の扱いの単純な方法は無視することであるが，未知語は意味的に重要な情報量を持っていることが多い. 例えば，商品名や地名などは企業への問い合わせでは重要な意味を持っている. 今回は [Kenter 15] を参考に，未知語ついてはランダムなベクトルを生成し，それを辞書として保持するようにした. もちろん, この方法は未知語の意味については何ら考慮されていないが，未知語の一致・不一致を扱えるぶん無視するよりは良いと考えられる.

4.2.5 意味上の特徴量について

最終的に, 文の分散表現としては次の 2 つを用いた. 式 8 により重み付けしたワードベクトルの行列 X に対して, 最小・最大・平均の畳み込みベクトルを連結した $SV^{mma\_idf} = [C^{min}; C^{max}; C^{avg}]$ と，絶対値の大きい要素で畳み込んだ $SV^{argmax\_abs\_idf} = C^{argmax\_abs}$ である.

そして，意味上の特徴量としては下記を用いた.

- SV^{mma_idf},$SV^{argmax\_abs\_idf}$ のコサイン類似度
- max(s_{i1} の未知語率,s_{i2} の未知語率)

未知語率を特徴量として加えるのは，未知語が相対的に多い場合には意味上の類似度よりも, 表記上の類似度を重視するようにモデルが学習することを狙ったものである. この効果については評価実験の節で述べる.

4.3 表記上の特徴量

表記上の特徴量としては下記を用いた.

- bag-of-words,bag-of-bigram,bag-of-trigram,bag-of-固有名詞のジャッカード係数
- 文 s_{i1} の長さ
- 文 s_{i1} の長さと文 s_{i2} の長さの比

ジャッカード係数は 2 つの集合の類似度してよく用いられる指標で,$J(A,B) = \frac{A \cap B}{A \cup B}$ で計算される.

5. 評価実験

5.1 データセット

今回，評価用のデータセットとして人手で作成した異なる 5 つのドメインでの FAQ 応答用の用例データベースを用いた. 概要を表 2 に載せる. 例えば，Test Set1 には 383 個の用例（質問と応答）が含まれており，応答の種類数としては 278 個のため，105 個の質問は他の質問を言い換えたものとなる. また，質問文の中に含まれる単語中の未知語（word2vec に含まれない) の割合を未知語率とし，データセット内の全質問の平均を平均未知語率として載せた.Test Set2 は未知語が少なく，Test Set5 は未知語が比較的多いデータセットだと分かる.

表 2: 5 つのテストセットの概要

Test Set	用例数（質問数）	応答種類数	平均未知語率
Test Set1	383	278	1.4%
Test Set2	428	208	0.7%
Test Set3	82	30	5.9%
Test Set4	176	60	3.6%
Test Set5	685	198	13.0%

5.2 評価方法

評価方法の概要を図 1 に載せる. まず，データセットをランダムに学習用と評価用に分割する. 学習用のデータセットを用いて FAQ 応答モデルを作成し，評価用のデータセットを用いて評価を行う. 評価用のデータセットに含まれる用例 < 質問 q',応答 r' > の質問 q' を FAQ 応答モデルに入力し，FAQ 応答モデルからスコアの高い順にランク付けされた上位 10 個の候補応答 < ランク i,応答 r_i,スコア $score_i$ > のリストを受け取る. そのとき，r' と r_i が一致する場合に正解とし下記の式で計算される平均逆順位 (MRR) によって評価する.

$$MRR = \frac{1}{N_Q} \sum_{i=1}^{N_Q} \frac{1}{fr_i} \quad (9)$$

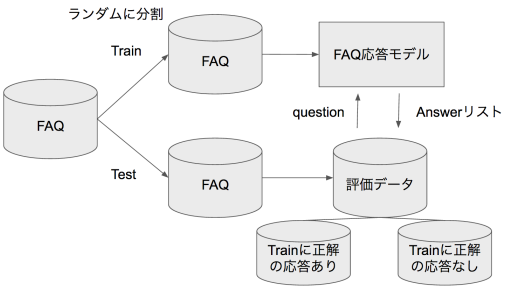

図 1: 評価方法の概要

ここで，fr_i は i 番目の評価データにおける最初に正解が出た順位であり,N_Q は評価データ数である. 期待する応答を上位に返せているほどスコアは高くなり，すべての質問に対して一番目に正解を返せていれば MRR は 1.0 となる.

また，FAQ 応答システムの実応用としては，間違いの可能性がある一つの応答を決定的に返すより，応答候補を複数提示して選択的にすることの方がユーザビリティとして望まれることがしばしばある. そこで，Top-N 正解率 (上位 N までに正解を含む評価データ数/全評価データ数 N_Q) も評価指標として加える. 今回は Top-3 正解率とした. なお，上記評価値の計算に用いた評価データは, 学習用のデータセットに正解の応答があるものに絞る.

上記を 5-fold cross-validation を用いて評価した.

5.3 ベースライン

今回は，ベースラインとして IBM Watson の日本語 Natural Language Classifier(NLC)[*1] を用いた.NLC は IBN Watson が提供するクラウド型の自然言語分類 API であり，自然文とラベルのペアからなる学習データを与えることで自然言語分類のモデルが学習され，自然文を入力することでラベルが信頼度と共に高い順に出力される. なお，実験時の version は v1 である.

5.4 セットアップ

本実験では，日本語形態素解析器として MeCab[*1] を用いており，辞書には mecab-ipadic-NEologd[*2] を使用した. また，抽出する品詞は名詞・動詞・形容詞・形容動詞のみとして，さらに非自立語は取り除いた.

単語の分散表現は word2vec(gensim[*3]) を用い，学習コーパスとして日本語 wikipedia[*4] を使用した.word2vec のモデルは skip-gram を用いて，主要なハイパーパラメータとしては，単語の次元数を 250，窓幅を 5，階層的ソフトマックスは使わず負例サンプリング数を 5 とし, 最小出現頻度は 40 とした.

また，予測モデルのトレーニングセットは，学習用の用例 DB から作成し，応答 r_i が一致する質問のペア $< q_{ij}, q_{ik} >$ を正例としラベルを 1.0, 一致しない質問のペア $< q_{ij}, q_{hk} >$ を負例としラベルを 0.0 とした. なお，負例は正例数に比べて圧倒的に多いので正例数の 50 倍にダウンサンプリングした. 特徴量は 4.2,4.3 節で説明したものを用い，予測モデルは予備実験の結果 Gradient Boosting Regression Tree(GBRT) を用いた.GBRT は決定木をベースとしたアンサンブルモデルであり, 特徴量の非線形な相互作用を自動で検出することが一つの特徴である. これにより, 未知語率に応じて意味上の特徴量と表記上の特徴量の寄与度を自動的にバランシングすることを期待する.

∗1 http://taku910.github.io/mecab/
∗2 https://github.com/neologd/mecab-ipadic-neologd
∗3 https://radimrehurek.com/gensim/models/word2vec.html
∗4 https://dumps.wikimedia.org/jawiki/latest/

表 3: 5 つのテストセットによる各手法の比較 (baseline との比率)

	Test Set1		Test Set2		Test Set3		Test Set4		Test Set5	
logic	MRR	Top3	MRR	Top3	MRR	Top3	MRR	Top3	MRR	Top3
Watson(baseline)	100.0%	**100.0%**	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	**100.0%**	**100.0%**
AVG_COS	99.1%	96.0%	97.7%	97.0%	97.1%	100.0%	104.2%	100.1%	79.5%	77.2%
MMA_IDF_COS	100.1%	96.7%	104.7%	105.1%	101.7%	104.0%	116.7%	112.0%	82.1%	81.4%
Proposed	96.7%	94.0%	104.0%	**107.1%**	103.3%	**112.0%**	112.5%	112.0%	84.1%	84.6%
Proposed(ALL)	**101.7%**	98.0%	**107.1%**	**107.1%**	**108.5%**	108.0%	**118.3%**	**117.1%**	91.6%	89.1%

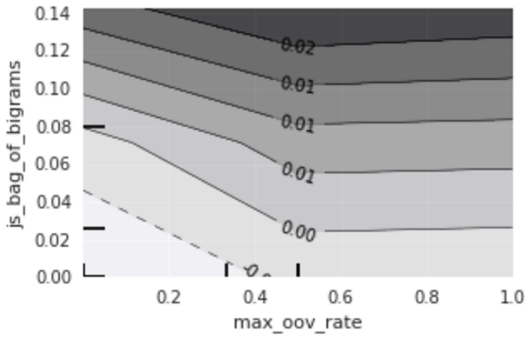

図 2: 未知語率 (横軸) × bag-of-bigram のジャッカード係数 (縦軸) の partial dependence

5.5 評価結果

評価結果を表 3 に載せる.AVG_COS は平均ベクトル (C^{avg}) のコサイン類似度,MMA_IDF_COS は SV^{mma_idf} のコサイン類似度, Proposed は提案手法であり各 Test Set ごとに予測モデルを作成した. また，Proposed(ALL) は 5 つの Test Set の学習データを混ぜて 1 つの予測モデルを作成し，それを各 Test Set の予測に用いた.

表 3 の通り，提案手法である Proposed(ALL) はベースラインと比較して,3 つのテストセットでより良い性能を示し, 1 つは同等, 1 つは悪い結果となった. 悪い結果となった Test Set5 は表 2 の通り, 他の 4 つの Test Set に比べて平均未知語率が高いことが特徴であり, このことが結果が悪い要因と考えられる.

また，Proposed と Proposed(ALL) の比較より，異なるドメインのデータを学習データとして混ぜても問題はなく, むしろ学習データ量が増えることで性能は改善した. このことは, 人間も, ドメインには非依存で 2 つの文の類似度をある程度判別できることからも自然な結果であると思われる.

図 2 に提案手法 Proposed(ALL) で学習した GBRT の特徴量の分析結果の一部を載せる. 図 2 は, 未知語率 (横軸) と bag-of-bigram のジャッカード係数 (縦軸) の特徴量の相互作用を可視化したものであり, 色が濃いエリアは予測への貢献が大きいことを示す. この図より, 未知語率が高いときには bag-of-bigram のジャッカード係数を重視していることが分かり, 未知語率を特徴量として加えることにより, 意味の特徴と表記の特徴の比重のバランシングに寄与していることが確認できる.

6. 考察・まとめ

本稿では，用例 DB からの応答選択について, 意味と表記の組み合わせによるモデルを提案し, 異なる 5 つのドメインの FAQ 用例 DB を使った評価実験を行い, ベンチマークとした IBM Watson の NLC より性能が上回ることを示した.5 つのうち,1 つのドメインについては劣る結果が確認されたが, この原因の一つは我々の word2vec における未知語率が高かったことが考えられる. 本実験では word2vec の学習用のコーパスとして, 日本語 wikipedia のみを用いたので, 用例 DB を word2vec の学習コーパスに加えたり, 未知語について近似ベクトルを推測する方法などを検討したい. また, 今回は学習データの類似度のラベルを 0 か 1 で与えたが, 中間表現を与える方法も検討したい. さらに, 今回は自社データを用いた独自評価のため, オープンなデータでの評価実験も今後のタスクである.

参考文献

[Vinyals 15] Vinyals, O., and Le, Q. 2015. A neural conversational model. Proceedings of the International Conference on Machine Learning, Deep Learning Workshop

[Lee 09] C. Lee, S. Lee, S. Jung, K. Kim, D. Lee and G.G. Lee, "Correlation-based query relaxation for example-based dialog modeling." Automatic Speech Recognition Understanding, 2009. ASRU 2009. IEEE Workshop on. IEEE, 2009.

[Lasguido 16] Lasguido, N. I. O., et al. "Neural Network Approaches to Dialog Response Retrieval and Generation." IEICE TRANSACTIONS on Information and Systems 99.10 (2016): 2508-2517.

[Mikolov 13] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.

[Pennington 14] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global Vectors for Word Representation." EMNLP. Vol. 14. 2014.

[Zhang 15] Zhang, Biao, et al. "Shallow Convolutional Neural Network for Implicit Discourse Relation Recognition." EMNLP. 2015.

[Kenter 15] Kenter, Tom, and Maarten de Rijke. "Short text similarity with word embeddings." Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. ACM, 2015.

[Kiros 15] Kiros, Ryan, et al. "Skip-thought vectors." Advances in neural information processing systems. 2015.